بانکداری الکترونیکی

امنیت فراهم کننده خدمات‌پرداخت

Payment Service Provider Security

مرکز ماهر

پیشگفتار

بدیهی است گسترش و ترویج پرداخت الکترونیکی مستلزم نصب، راه اندازی، آموزش و نگهداری پایانه های فروش (POS) به صورت گسترده در محل های فروش کالا و خدمات، و همچنین راه اندازی امکانات نرم افزاری برای پرداخت از طریق اینترنت و موبایل است. طبق مصوبه پول الکترونیکی، بانک های کشور می توانند پس از انعقاد قرارداد با شرکت های ارایه دهنده خدمات پرداخت (PSP) جهت ارایه خدمات فوق که دارای مجوز بانک مرکزی جمهوری اسلامی ایران باشند، وظیفه ارایه خدمات پرداخت را به شرکت های PSP محول نمایند.

طبق تعریف بالا شرکت های ارائه کننده خدمات پرداخت یا Payment Service Provider ها که از آنها به اختصار با عنوان PSP یاد می کنیم؛ شرکت هایی هستند که دارای مجوز از بانک مرکزی جمهوری اسلامی ایران بوده و ضمن عقود قرارداد با بانک ها و موسسات مالی و اعتباری می توانند خدمات پرداخت را به آن موسسات یا مشتریان آنها ارائه نمایند. به طور خلاصه تر وظیفه اصلی این شرکت ها فراهم ساختن سخت افزار، نرم افزار و تجهیزات مورد نیاز برای ایجاد شبکه ای است که پردازش تراکنش پرداخت دارنده کارت به پذیرنده را میسر سازد.

در این گزارش هدف بررسی آناتومی خدمات پرداخت، تهدیدها و آسیب‌های موجود در این خدمات و مکانیزم‌های دفاعی ممکن می‌باشد.

فهرست مطالب

فهرست جداول

فهرست اشکال

فصل اول:

فرآیند تراکنش‌های پرداخت

به‌منظور درک نقاط ضعف برنامه‌های پرداخت و پایانه فروش[۱]، لازم است که بفهمیم چطور و چه‌موقع و کجا داده‌های صاحبان‌کارت[۲] بین دو نقطه متفاوت در چرخه تراکنش جابه‌جا می‌شود.

کارت‌های پرداخت

نحوه استفاده از کارت‌های پرداخت موضوع اصلی مورد بحث در این کتاب است. انواع مختلفی از کارت‌های پرداخت بطور عمومی در پرداخت‌ها مورد استفاده قرار می‌گیرد:

کارت اعتباری[۳] اولین نوع از کارت‌های پرداخت می‌باشد که هنوز هم خیلی فراگیر است. با پرداخت از طریق کارت اعتباری، مشتریان از اعتبارات در دسترس خود استفاده کرده و قبض‌ها را پرداخت می‌کنند. کارت‌های اعتباری معمولا توسط PIN[۴] بطور کامل محافظت نمی‌شود تا به مشتریان اجازه خرید برخط را بدهد.

کارت بدهی[۵] (کارت خدمات بانکی عادی) (ATM, پول نقد) روش نسبتا جدیدی برای پرداخت است. کارت بانکی با کارت اعتباری متفاوت است چون صاحبان کارت بانکی با پول در دسترس در حساب‌های بانکی خود پرداخت می‌کنند، که به‌سرعت این کسر حساب صورت می‌گیرد. به‌نظر می‌رسد که کارت بانکی در مقایسه با کارت اعتباری خطرناک‌تر باشد چون کارت بانکی مستقیما به حساب بانکی متصل است و معمولا به ATM اجازه برداشت پول نقد را می‌دهد. از طرفی دیگر، با استفاده از صحت‌سنجی دو مرحله‌ای (شماره رمز و خود کارت) خیلی ایمن‌تر می‌شود. یکی از مولفه‌های پرخطر برای هر کارت بانکی این است که با وارد نکردن رمز می‌تواند بعنوان کارت اعتباری مورد استفاده قرار بگیرد.

کارت هدیه مشابه کارت بانکی عادی است اما معمولا رمز ندارد. کارت هدیه به حساب بانکی متصل نیست و معمولا شامل مقدار مشخص و ثابت پول است. در خود کارت هیچ اطلاعات مالی وجود ندارد و فقط پایانه فروش با ارائه دهندگان کارت هدیه تبادل اطلاعات و صحت‌سنجی می‌کنند. کارت‌های هدیه نسبت به دو کارت قبلی کم خطرتر هست چون مقدار ثابت و معین شده از پول را شامل می‌شود.

---

[۱] Point-Of-Sale (POS)
[۲] cardholder
[۳] Credit card
[۴] Personal Identification Number
[۵] Debit card

کارت ناوگانی[۶] (سوخت) شبیه به کارت اعتباری است ولی باید در جاهای مخصوصی استفاده شود (معمولا جایگاهای گاز و فروشگاه‌های بزرگ) و برای خرید فقط به کالاهای معدودی محدود می‌شود (مانند سوخت و دیگر کالاهای خودرو). کارت‌های ناوگانی، با اینکه با بیشتر برندهای ارائه دهندگان کارت مشکل دارد ولی به ندرت مورد توجه کلاهبرداران قرار می‌گیرد چون این کارت برای برداشت‌های ATM، خرید برخط، یا خریدهای مهم در فروشگاه‌ها مورد استفاده قرار نمی‌گیرد.

جدول ۱-۱: لیستی از انواع مختلف کارت و ویژگی‌های اصلی آن‌ها را نشان می‌دهد.

نوع کارت	صادر شده از	قدرت خرید, $$	مقبولیت	محافظت شده با استاندارد PCI ؟
اعتباری	توسط بانک‌های تحت برندهای پرداخت (مانند ویزا) یا مستقیما توسط برندهای پرداخت (American Express)	چندین هزار	به‌طور مجازی هر تاجر برخط	بله
بانکی	توسط بانک‌های با برند یا بدون برند	چندین هزار	بطور مجازی هر تاجر؛ ATM بانک	فقط در صورت سازگاری با برند پرداخت
هدیه	توسط برندهای پرداخت یا ارائه دهندگان اختصاصی	چند صد	اگر معتبر باشد، به‌طور مجازی هر تاجر برخط. اگر اختصاصی باشد، فقط تاجران خاص.	فقط در صورت سازگاری با برند پرداخت
ناوگانی	توسط بانک‌ها، برندهای پرداخت یا ارائه دهندگان اختصاصی	چند صد	تاجران خاص و انواع مختلف و محدودی از کالا	فقط در صورت سازگاری با برند پرداخت

روش‌های ورود کارت

دو روش عمومی از طریق کشیدن کارت و دستی برای ورود اطلاعات به پایانه فروش به‌منظور شروع تراکنش پرداخت وجود دارد.

MSR

[۶] Fleet card

روش اول از نوارمغناطیس‌خوان[۷] استفاده می‌کند، دستگاهی که نوار مغناطیس بر روی کارت‌های پرداخت را می‌خواند. دستگاه‌های مدرن نوارمغناطیس‌خوان قابلیت رمزنگاری را دارد و در رمزنگاری‌های نقطه به نقطه[۸] مورد استفاده قرار می‌گیرد. راحت‌ترین روش برای وارد کردن اطلاعات کارت به پایانه فروش این است که فقط کارت را به دستگاه نوارمغناطیس‌خوان بکشید تا با خواندن اطلاعات از طریق نوار مغناطیسی تمام اطلاعات ضروری مورد نیاز را به‌طور خودکار وارد کند. هرچند که، اگر نوار مغناطیس معیوب باشد، مشتری می‌تواند با وارد کردن دستی شماره و تاریخ انقضای کارت فرایند خرید را انجام دهد.

بسیاری از دستگاهای نوارمغناطیس‌خوان صفحه کلید ورودی را شبیه سازی می‌کنند، بنابراین عملیات کشیدن کارت معادل با وارد کردن اطلاعات از طریق صفحه کلید رایانه می‌باشد. دزدیدن مسیر[۹] داده در این مورد، به راحتی شنود[۱۰] اطلاعات ورودی نوارمغناطیس‌خوان با نصب ضبط کننده صفحه کلید[۱۱] است.

Pinpad

دومین روش، استفاده از صفحه‌رمز[۱۲] است. صفحه‌رمز یا نقطه تبادل[۱۳] همراه با نوارمغناطیس‌خوان، به دلیل داشتن سخت‌افزار قابل تنظیم برای کارایی مختلف مانند محافظت از اطلاعات حساس کارت، یک دستگاه به نسبت پیچیده‌تری است. بیشتر صفحه‌رمزها، رمزگذاری سخت‌افزاری دارند که با نام ماژول امنیتی مقاوم در برابر حمله[۱۴] یا TRSM شناخته می‌شود. برای تبادل اطلاعات بهتر با مشتری در طول فرآیند پرداخت علاوه بر نوارمغناطیس‌خوان و پایانه فروش امکانات جانبی از قبیل صفحه کلید در دسترس مشتری (علاوه بر صفحه‌رمز) قرار دارند.

عوامل اصلی

بر اساس گفته‌های شرکت ویزا، پنج عامل اصلی مصرف‌کنندگان، تاجران، پذیرندگان، صادرکنندگان و برندهای کارت در فرآیند پرداخت نقش کلیدی دارند. هرچندکه، در عمل عوامل دیگری از قبیل درگاه، پردازشگرها[۱۵]، فروشندگان نرم‌افزار و تولیدکنندگان سخت افزار هم وجود دارند که فرایند تراکنش پرداخت را تمهید می‌کنند.

---

۷ Magnetic Stripe Reader (MSR)
۸ Point-to-Point encryption (P2PE)
۹ track
۱۰ Sniffing
۱۱ Keystroke logger
۱۲ Pinpad
۱۳ Point Of Interaction (POI)
۱۴ Tamper-Resistant Security Module
۱۵ Processors

قبل از ادامه بحث مربوط به عوامل اصلی، لازم است یادآور شویم که برخلاف این واقعیت که تاجران نقش کم‌رنگی در فرآیندهای پرداخت دارند ولی مسئولیت و خطرات بسیار زیادی نسبت به عوامل دیگر متقبل می‌شوند. از جمله این خطرات می‌توان به موارد زیر اشاره کرد:

۱. تاجران نسبت به عوامل دیگر ساختار گسترده‌تری دارند، به صورتی که یک زنجیره خرده فروشی شامل هزاران فروشگاه می‌شود. این گفته را با پردازنده‌هایی که چند مرکز داده در مقیاس نهایی دارند و کنترل و سازماندهی آن خیلی راحت‌تر است مقایسه کنید.
۲. فروشگاه‌های خرده فروشی مکان‌های عمومی هستند که تمام مسائل مربوط به امنیت را شامل می‌شوند.
۳. بیشتر تاجران به فروشندگان نرم‌افزاری و سخت‌افزاری که ارائه دهندگان تکنولوژی (شامل امنیت) می‌باشند، تکیه دارند و به‌راحتی آماده پذیرش این مسئله نیستند که روش موجود از باب طراحی آسیب‌پذیر است. وقتی انقلاب رایانه و اینترنت در اواخر دهه ۹۰ میلادی شروع به جایگزینی روش‌های قدیمی ثبت پول و پایانه‌های اعتباری مستقل با سیستم‌های پیشرفته پایانه فروش همراه با برنامه‌های پرداخت کرد، و تعداد نامحدودی از آسیب‌های امنیتی سیستم و شبکه با خود به همراه آورد پس به تدریج این آسیب‌ها تبدیل به کابوس روز افزون برای خرده فروشان سرتاسر جهان شد.

مصرف‌کننده (صاحبان کارت)

به‌طور پیش‌فرض، مصرف‌کنندگان قرار نیست درمورد حفظ رمز نگران باشند. اگر کارت گم بشود مصرف کننده فقط کافی است با بانک تماس بگیرد تا کارت جدید تهیه نماید. وقتی کارت ما ، از طریق سیستم‌های پایانه فروش که قرار است از اطلاعات ما در طول فرآیند حفاظت کند کشیده می‌شود، اطلاعات شخصی ما با تاجران به اشتراک گذاشته می‌شود. پس ما به تکنولوژی‌های بسیار مدرن امروزی متکی هستیم تا از پول کارتی خود محافظت کنیم.

اما در عمل، متاسفانه، این اتفاق رخ نمی‌دهد. همه کارت‌ها توسط رمز محافظت نمی‌شوند. بنابراین، اگر کارت گم یا دزدیده شود، این حقیقت نادیده قرار گرفته می‌شود که ممکن است پول مصرف‌کننده دزدیده شود. وقتی کارت در پایانه فروش کشیده می‌شود، اطلاعات بطور کاملا محرمانه نگهداری نمی‌شود، پس در صورتی که فیش ماهانه مبلغ قابل توجهی را نشان بدهد، مسئله تعجب‌برانگیزی نخواهد بود.

تاجر

تاجران، مانند سوپرمارکت‌ها، فروشگاه‌های بزرگ، رستوران‌ها یا هتل‌ها عوامل مرکزی در فرآیند پرداخت هستند. این عوامل تصمیمات مختلف تجاری و فنی می‌گیرند، اینکه چه نوع پرداخت‌هایی باید پذیرفته شود یعنی عتباری یا بانکی، یا هردو؛ چه برندهایی باید پذیرفته شود، چه بانکی باید در آن حساب باز کرد، چه

پایانه فروش و چه سخت‌افزار و نرم‌افزاری برای خرید باید استفاده کرد؛ و در آخر چگونه از اطلاعات خریدار محافظت شود. این تصمیم آخر ممکن است با بقیه متفاوت به نظر برسد و در مقایسه با تصمیمات دیگر نامربوط باشد ولی واقعیت این است که تاجران به دلیل عدم موفقیت عوامل اصلی در این زمینه محافظت باید امنیت اطلاعات پرداخت را مورد توجه قرار بدهند.

ناگفته مشخص است که، تاجران هنوز برای فروش کالاها و سرویس‌های خود از پرداخت‌های کارتی استفاده می‌کنند. سخت‌افزار و نرم‌افزارهای پایانه فروش آن‌ها اطلاعات کارت را قبول کرده و پردازش می‌کند، این اطلاعات را برای صحت‌سنجی به پردازش‌گرها ارسال می‌کنند و به‌تدریج پول را از حساب تاجر دریافت می‌کنند.

پذیرنده

پذیرندگان، بانک‌های پذیرنده، تراکنش پرداخت را صحت‌سنجی کرده و با صادرکنندگان کارت مطابقت می‌دهند. به‌منظور پاسخ به پذیرنده برای صحت‌سنجی، مسیر فرآیند تراکنش پرداخت بر اساس نوع کارت و تراکنش معین می‌شود. پذیرندگان نسبت‌های کاسته شده تاجر را تنظیم می‌کنند (فیش‌هایی که تاجر برای هر یک از فرآیند تراکنش پرداخت پرداخته است).

صادرکننده

صادر کنندگان، یا بانک‌های صادر کننده، حساب‌های مشتریان را نگه داشته و برای مشتریان کارت صادر می‌کند. آنها با توجه به تراکنش‌ها و پول‌های برگشتی به خریداران از مشتریان خود هزینه دریافت کرده و به تاجران پول پرداخت می‌کنند. صادرکنندگان کارت‌ها را تولید می‌کنند، پس آن‌ها مسئول حفاظت فیزیکی کارت‌ها هستند.

برندهای کارت

برندهای کارت یا شبکه‌های کارتی، تمام فرآیند صحت‌سنجی پرداخت و توافق را فراهم می‌کنند. شبکه‌هایی از قبیل VisaNet ارتباط مابین پذیرندگان و صادرکنندگان را برقرار می‌سازد. بسیاری از برندهای کارت مانند Visa و MasterCard، در فرآیند خرید و صدور کارت به‌طور مستقیم مداخله نمی‌کنند و این کارها را به سازمان سوم شخص[۱۶] مستقل می‌سپارند. برندهای دیگر از قبیل American Express، کارت را خودشان صادر کرده و تراکنش‌های پرداخت را انجام می‌دهند.

[۱۶] Third-party

عوامل اصلی دیگر

علاوه بر عوامل اصلی گفته شده در فرایند پرداخت، عاملی تحت عنوان "مرد میانی[۱۷]" وجود دارد که مقادیر زیادی از "اضافات" را برای تاجران تامین می‌کند. بطور نظری، تاجر ممکن است که با ارتباط مستقیم با پذیرندگان، پرداخت‌های الکترونیکی را بدون این سازمان اضافی قبول کند. در عمل، هرچند که، پیچیدگی طرح‌های مربوط به فرآیند پرداخت و مبالغ بالای کارت‌ها و روش‌های پرداخت، چنین اموری بدون مداخله تکمیل‌کنندگان و درگاه‌ها تقریبا غیرممکن است.

پردازشگر پرداخت

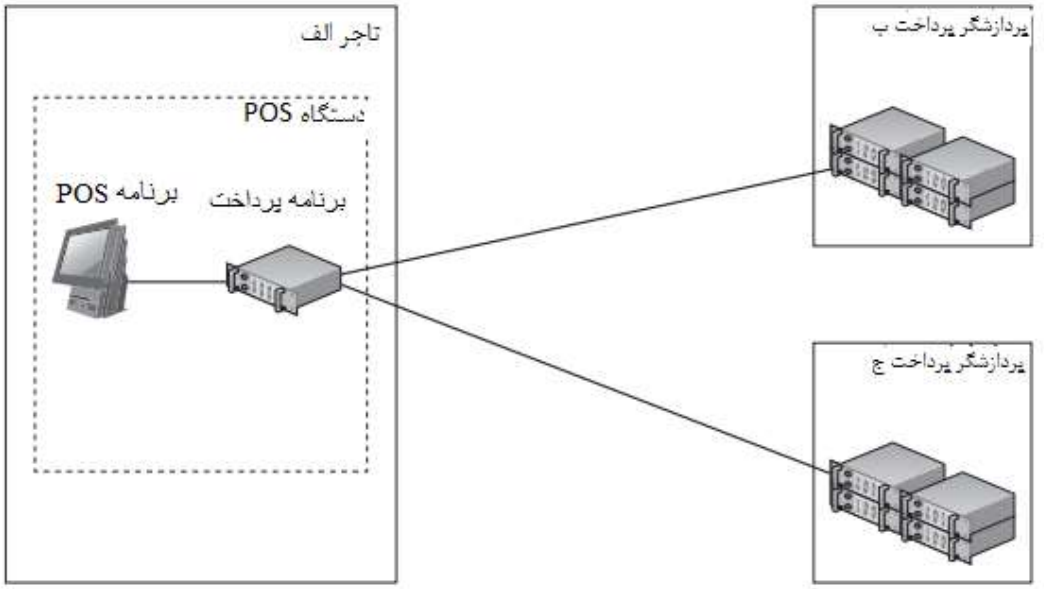
پردازشگر پرداخت، تراکنش‌های پرداخت را مابین تاجر و چندین پذیرنده کنترل می‌کند. همچنین حساب‌های تاجران را در جائیکه که تاجران به‌طور واقعی پول‌های پرداختی خود را از صاحبان کارت برای کالاها یا خدمات دریافت می‌کنند را نگه می‌دارند.

پردازشگرها، تراکنش‌های پرداخت پذیرندگان را بر اساس نوع پرداخت و برند کارت هدایت می‌کند. پردازشگرهای پرداخت برای فرآیندهای مختلف، گزارشات مالی فراهم می‌کنند. عملکردهای مفید دیگری نیز پردازشگرهای پرداخت ارائه می‌دهند اما در بسیاری از موارد نمی‌توانند امنیت داده‌های پرداخت تاجران را به‌راحتی تامین کنند چون از منابع و ذخایر آن‌ها اطلاعی ندارند.

پردازشگرها ممکن است که مکانیزم‌های اضافی دیگر از جمله رمزنگاری نقطه به نقطه و تقسیم‌بندی[۱۸] را پیشنهاد بدهند. هرچند، زمانی‌که بسیاری از تاجران از نرم‌افزار و سخت‌افزارهای سوم‌شخص برای پشتیبانی بیش از یک پردازشگر پرداخت استفاده می‌کنند این پیشنهادها اغلب مشکلات امنیتی را بطور کامل حل نمی‌کند. علاوه بر این، ویژگی‌های تقسیم‌بندی ارائه شده توسط بسیاری از پردازشگرها، مشکل امنیت اطلاعات کارت را حل نمی‌کند.

در مثال نشان داده شده در شکل ۱-۱، ممکن است تاجر تمام فرآیند تراکنش‌های اعتباری را با پردازشگر **ب** انجام بدهد، اما به پردازشگر **ج** تراکنش‌های کارت هدیه را ارسال کند.

[۱۷] Man-in-the-middle
[۱۸] Tokenization



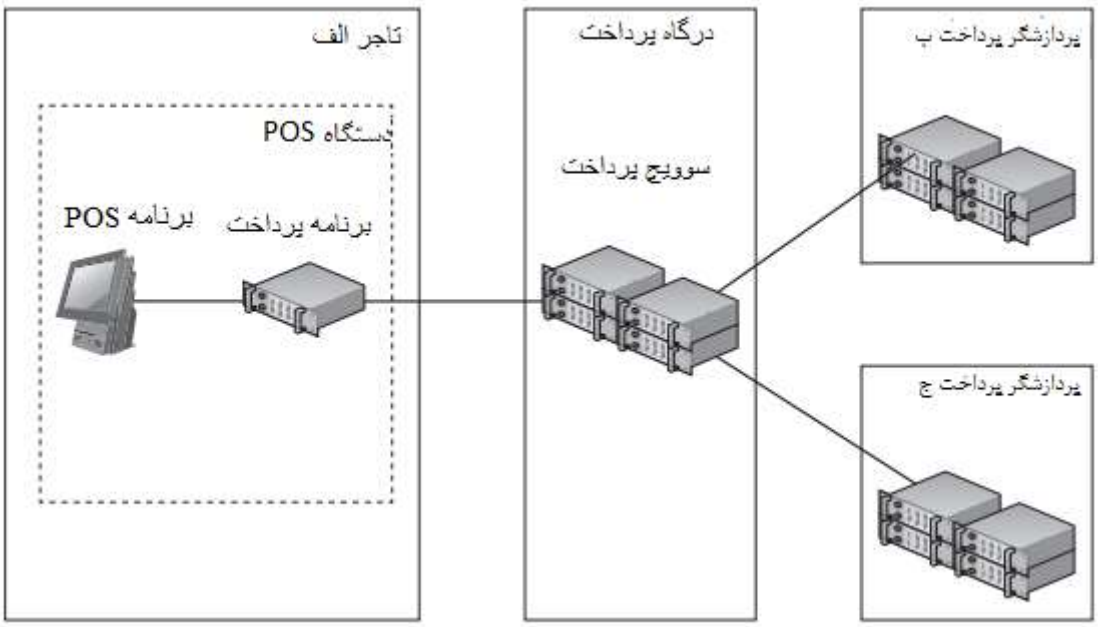
شکل ۱-۱: تاجر متصل شده به تکمیل‌کننده پرداخت

برخلاف پذیرندگان، پردازشگرهای پرداخت انواع مختلفی از کارت‌ها و روش‌های پرداخت از قبیل، کارت‌های هدیه، کارت‌های ناوگانی، انتقالات سودمند الکترونیکی و سایر روش‌ها را پشتیبانی می‌کنند. و برای کارت‌های اعتباری و بانکی محدودیتی وجود ندارد.

درگاه پرداخت

در بسیاری از موارد، سیستم پرداخت پایانه فروش تاجر به‌طور مستقیم با پردازشگر پرداخت صحبت می‌کند. بعضی مواقع، هرچند، مابین تاجر و پردازشگر پرداخت یک واسط با نام درگاه پرداخت (یا کلید پرداخت) وجود دارد که وظیفه اصلی آن ارائه خدمات درگاه یا راهنمایی به تاجران است.

موقعیتی را تصور کنید که وقتی تاجر **الف** با پردازشگر **ب** توافق خدمات انجام داده‌اند که هزینه آن ۰٫۳ دلار به‌اضافه دو درصد هزینه برای هر تراکنش باشد. همه‌چیز بسیار عالی است تا زمانی‌که تاجر تبلیغاتی را برای پردازشگر پرداخت **ج** با هزینه ۰٫۲۹ دلار و ۱٫۹ درصد برای هر تراکنش را مشاهده می‌کند. این تفاوت در ظاهر بسیار کم ممکن است برای تاجر الف که هزاران تراکنش انجام می‌دهد، پول بسیار زیادی را صرفه‌جویی کند. هرچند، با تغییر پردازشگر پرداخت از **ب** به **ج**، تاجر **الف** باید به فروشنده نرم‌افزار پایانه فروش مبلغ ۲۰۰ هزار دلار به دلیل تغییر در برنامه پرداخت کند تا بتواند با پردازشگر **ج** ارتباط برقرار کند؛ چون این نرم‌افزار ذاتا طراحی شده است تا با پردازشگر **ب** کار کند. شکل ۱-۲ نشان می‌دهد که اگر تاجر **الف** از درگاه پرداخت استفاده می‌کرد، چنین تغییراتی در نرم‌افزار پایانه فروش ظاهر می‌شد چون ممکن بود در سوویچ درگاه پرداخت، مسیریابی انجام گیرد که از داده مرکزی نشات می‌گیرد.



شکل۱-۲: تاجر متصل شده به تکمیل‌کننده پرداخت

درگاه‌های پرداخت ممکن است که سرویس‌هایی با سازگاری بیشتری از قبیل رمزنگاری نقطه به نقطه، گزارش متمرکز از مدیریت دستگاه پایانه فروش، و تقسیم‌بندی را ارائه بدهند.

تفاوت اساسی مابین پردازشگر پرداخت و درگاه در این است که پردازشگرها، علاوه بر عملکرد سوییچ ارائه شده توسط درگاه‌ها، همچنین حساب‌های تاجران را حفظ کرده و فرآیند توافق را فراهم می‌کنند.

نقش مهم دیگری که درگاه باید ایفا کند این است که نرم‌افزاری را ارائه دهد که بتواند اجرا شود و در نهایت با برنامه پرداخت/پایانه فروش و در طرف دیگر با سوییچ درگاه (سرور در مرکز داده در حال اجراست) صحبت کند. بدین ترتیب، درگاه پرداخت، ممکن است امنیت ساختار پرداخت تاجر را متاثر کند. که این تاثیر ممکن است آن‌را بهبود بخشد (برای مثال، با ارائه عملکرد P2PE) یا می‌تواند به آن صدمه وارد کند (بوسیله جایگذاری مولفه‌های معیوب خود در سیستم پایانه فروش ایمن قبلی). در بیشتر مواقع، تاجر هنوز هم مسئول امنیت درگاه برنامه اجرا شده مشتری در فروشگاه است.

عوامل بیشتر

هرچندکه عوامل اصلی از قبیل صادرکنندگان، پذیرندگان، و برندهای کارت، مولفه‌های ضروری در فرآیند پرداخت به شمار می‌آیند، واقعیت این است که تاثیر بسیار کمی بر امنیت اطلاعات پرداخت در فروشگاه‌های تاجر دارند. دلیل این امر این است که به دور از فضای نا امن خرده فروشی، نقش آن‌ها فقط در مواقع نیاز پررنگ می‌شود. پردازشگرها و درگاه‌ها کمی بیشتر به واقعیت نزدیک هستند چون مستقیما با فروشگاه‌ها در ارتباط هستند و بعضی مواقع نرم‌افزار مورد اجرای پایانه فروش را تامین می‌کنند. هرچندکه، بر موقعیت

فعلی هیچ کنترلی ندارند چون به رابط‌های آن‌ها در محیط پرداخت فقط کمی پیچیدگی افزوده شده است، و عوامل دیگری نیز مانند فروشندگان نرم‌افزار پرداخت و تولیدکنندگان سخت افزار در فرآیند پردازش وجود دارد.

فروشندگان نرم‌افزار پرداخت

فروشندگان نرم‌افزار سوم‌شخص، پایانه فروش و برنامه‌های پرداخت را برای تاجران توسعه داده‌اند. این برنامه‌ها اطلاعات حساس را در کل فرآیند پرداخت، از لحظه کشیدن کارت تا برقراری ارتباط با سرور در فروشگاه خرده فروشی کنترل (پردازش، انتقال، و ذخیره) می‌کنند. اگر به بروشورهای اطلاعاتی ارائه شده توسط برندهای کارت مراجعه کنید، متوجه می‌شوید که فروشندگان نرم‌افزار در لیست عوامل فرایند پرداخت حضور دارند که این امر اشتباه است. فروشندگان نرم‌افزار برنامه‌هایی را تولید کرده‌اند که در فروشگاه‌ها نصب می‌شود، و در مابین چیزهای دیگر، قرار است که از اطلاعات صاحبان کارت محافظت کنند. برخلاف تاجران، توسعه دهندگان برنامه در موقعیت مناسبی برای به وجود آوردن و پیاده‌سازی تکنولوژی‌های پیچیده قرار دارند. اگر سیستم پایانه فروش، که معمولا توسط فروشندگان نرم‌افزار سوم‌شخص ساخته می‌شود و نه توسط خود تاجران، در تامین امنیت صاحبان کارت ناموفق باشد، کل فرآیند پرداخت با شکست مواجه خواهد شد چون فروشگاه‌های خرده فروش، هدف اصلی حملات مهاجمین خواهد بود.

برنامه پرداخت فروشندگان، مستلزم پیروی از استاندارد امنیت داده‌های برنامه پرداخت[۱۹] است، که در محافظت از داده‌های حساس به اندازه کافی قوی و موثر نیست.

تولیدکنندگان سخت‌افزار

تولیدکنندگان سخت‌افزار نمونه دیگری از عوامل اشاره نشده چرخه فرآیند پرداخت است. آن‌ها دستگاهای جانبی ضروری از قبیل نوارمغناطیس‌خوان و صفحه‌رمز را برای فرآیند پرداخت مهیا می‌کنند، این لوازم در خط مقدم امنیت داده‌های پرداخت قرار دارند چون ابتدا داده‌های صحت‌سنجی را با کشیدن نوار مغناطیسی یا ورود دستی قبول، پردازش، و منتقل می‌کنند. لوازم صفحه‌رمز باید با PCI PTS سازگار باشند تا بتوانند اجازه پردازش تراکنش‌ها را داشته باشند. هر دو لوازم نوارمغناطیس‌خوان و صفحه‌رمز باید مجهز به PCI PTS باشند تا بتوانند خدمات P2PE را ارائه بدهند. به همین دلایل، در توضیح پروسه پرداخت اغلب به تولیدکنندگان آن‌ها اشاره نمی‌شود.

[۱۹] Payment Application Data Security (PCI PA-DSS)

مراحل پرداخت

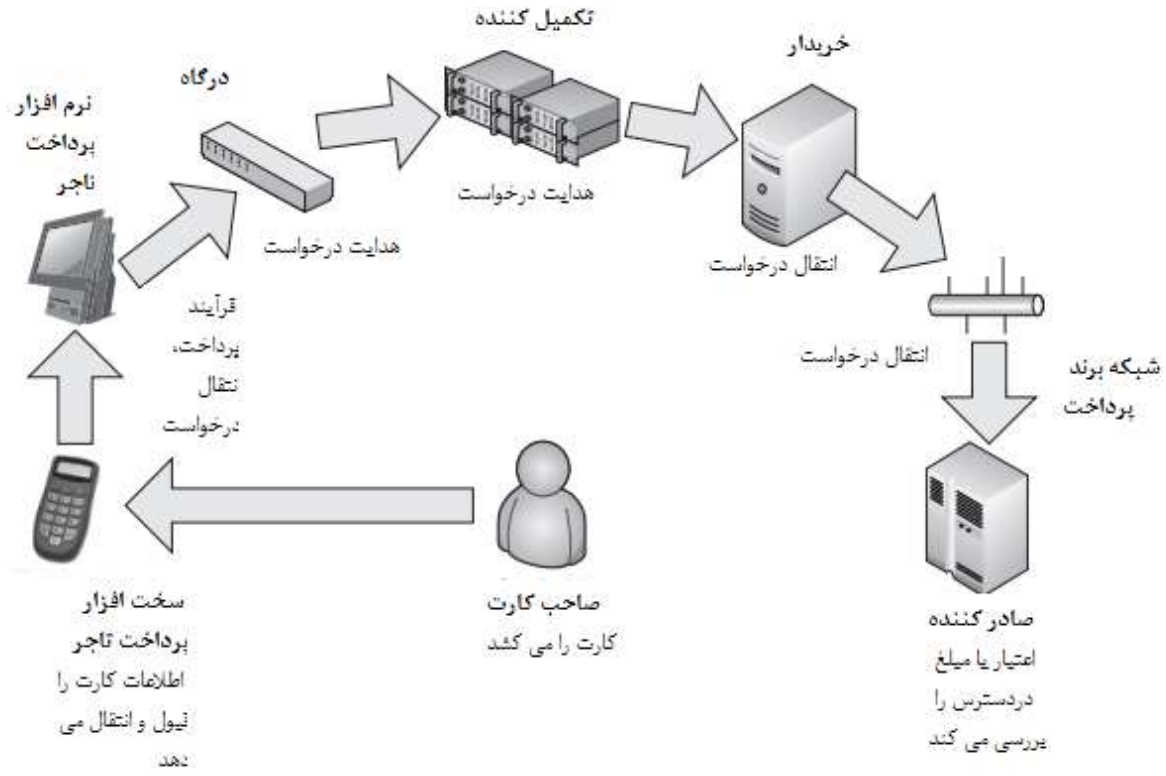
تراکنش پرداخت کارت از طریق مراحلی صورت می‌گیرد که ابتدا با کشیدن کارت در پایانه فروش شروع شده و با ارسال فیش به صاحبان کارت خاتمه می‌یابد. از نظر سیستم پرداخت، دو مرحله اصلی صحت‌سنجی و تایید ارتباط با سرور پرداخت وجود دارد. با توجه به برنامه پرداخت و پایانه فروش، در پشت این دو مرحله فازهای دیگری نیز وجود خواهد داشت.

صحت‌سنجی

اولین گام در فرآیند پرداخت صحت‌سنجی می‌باشد. به منظور بررسی کارت‌های اعتباری یا بانکی صاحبان کارت، این گام لازم است تا حساب بانکی صاحب کارت را برای تایید مبلغ کافی برای ادامه فرآیند پرداخت بررسی کنند. پروسه صحت‌سنجی در شکل ۱-۳ نشان داده شده است.

برای نمونه، در فروشگاه‌های خرده فروشی وقتی‌که حسابدار لوازم انتخاب شده توسط مشتری را با بارکد خوان اسکن می‌کند، حسابدار با فشردن کلید جمع کل مبلغ، به حالت پرداخت پایانه فروش متصل می‌شود. مشتری برای انتخاب یکی از روش‌های پرداخت هدایت شده و با توجه به کارت خود که ممکن است اعتباری، بانکی، هدیه یا EBT باشد کارت خود را در کارت‌خوان می‌کشد، این کارت‌خوان ممکن است که یک نوارمغناطیس‌خوان ساده یا یک دستگاه نقطه تبادل پیچیده باشد. برنامه‌های پرداخت اطلاعات کارت را تحلیل کرده و با توجه به نوع کارت و شماره شناسایی بانک[۲۰]، عملیات تراکنش را شروع می‌کند. گام بعدی برای تراکنش، درگاه پرداخت یا پردازشگر پرداخت است. داده‌های تراکنش به یک پذیرنده معتبر هدایت می‌شود تا بعدا به‌منظور دریافت تاییدیه با صادر کننده ارتباط برقرار کند. صادرکننده کسی است که اطلاعاتی را در مورد حساب صاحبان کارت نگه داشته و وضعیت آن‌را به صورت آنی بررسی می‌کند.

[۲۰] Bank Identification Number (BIN)



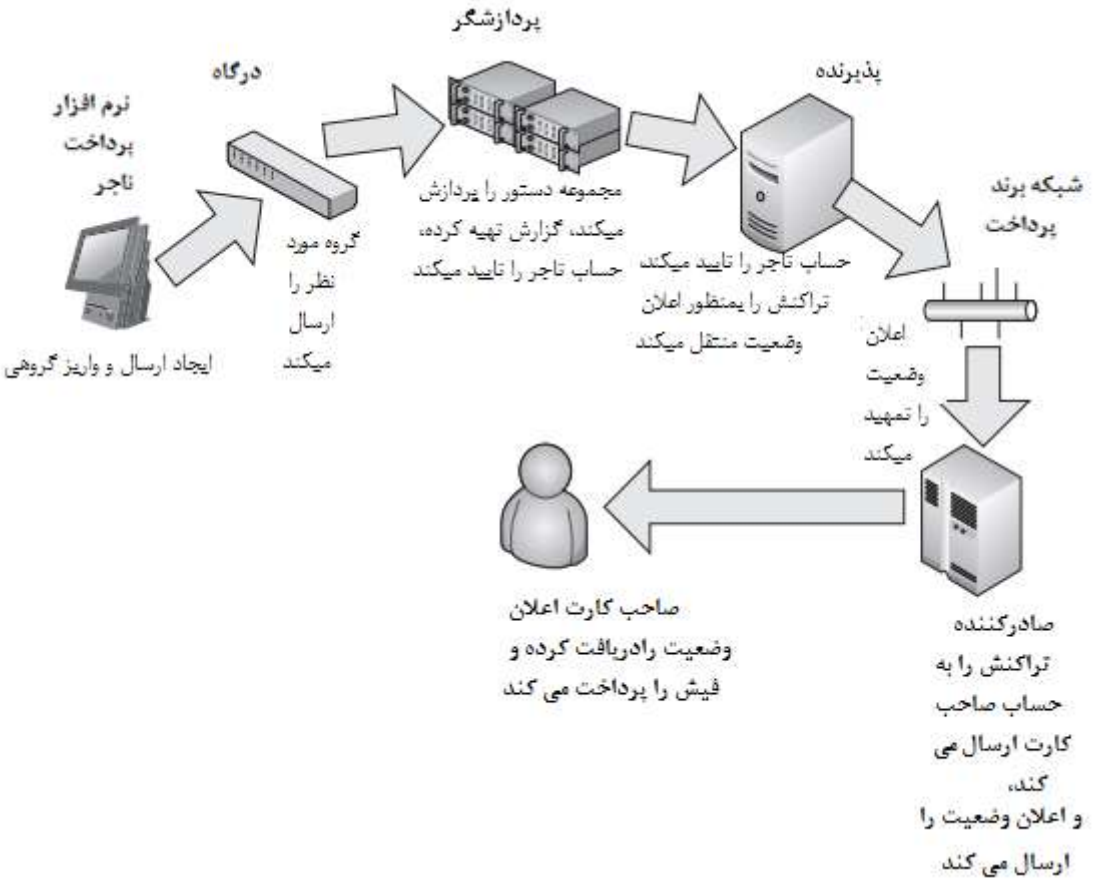
شکل ۱-۳: نمودار صحت‌سنجی

اگر مبلغ تراکنش تایید شود، صادرکننده مبلغ را بررسی کرده و با مبلغ تراکنش مقایسه می‌کند. اگر مشتری مبلغ کافی برای تامین مقدار تراکنش داشته باشد، صادرکننده یک تاییدیه‌ای به خریداری که آن‌را به پردازشگر پرداخت برگردانده است، ارسال می‌کند و دوباره به پایانه فروش برمی‌گردد.. در مواردی که کارت بانکی مطرح است، صادرکننده حساب بانکی صاحب کارت را به‌منظور تایید مبلغ کافی بررسی می‌کند. یک بررسی مشابه با یک اختلاف در اینکه کارت هدیه هیچ حساب بانکی در رابطه با کارت ندارد وجود خواهد داشت. در عوض، هر کارت هدیه به یک پایگاه داده خاصی متصل است که توسط ارائه دهندگان کارت هدیه نگه داری می‌شود. در هر شرایطی، اگر مشتری مبلغ کافی در حساب بانکی یا کارت هدیه خود نداشته باشد، تراکنش با خطا مواجه می‌شود، و این خطا به پایانه فروش برگشت داده می‌شود که یک پیام خطایی بر روی دستگاه کارت‌خوان مبنی بر استفاده از روش پرداخت دیگر ظاهر می‌گردد.

از نظر امنیت داده، مرحله صحت‌سنجی مهم‌ترین بخش است چون این مرحله نیازمند ارسال تمام اطلاعات حساس صحت‌سنجی (مسیر ۱ یا مسیر ۲ یا هردو) از طریق پایانه فروش به تمام سیستم پذیرنده است.

توافق[۲۱]

زمانی‌که صحت‌سنجی صورت گرفته و تراکنش توسط پایانه فروش نهایی شد، پرداخت باید توافق شود، که بدین معنی است که مابین تاجر، پذیرنده، و صادرکننده پرداخت توافق صورت گرفته است. در حین توافق، تاجر (به طور دقیق‌تر سیستم پرداخت مربوطه) اطاعات تراکنش را به پردازشگر ارسال کرده و سپس به پذیرنده ارسال می‌کند. پذیرنده یا پردازشگر، حساب صاحب کارت را تایید می‌کند، و داده را به صادرکننده که منتقل کننده تراکنش به حساب صاحب کارت است، ارسال می‌کند. شکل ۱-۴ نمودار توافق را نشان می‌دهد.



شکل ۱-۴: نمودار توافق

از نظر امنیتی، توافق نسبت به صحت‌سنجی کم‌خطرتر است چون با تمام داده‌های صحت‌سنجی حساس سروکار ندارد. زمانی‌که داده‌های صحت‌سنجی همزمان با دریافت اطلاعات از پایانه فروش (جدول ۱-۲) از بین می‌روند فرآیند توافق نیازمند جمع‌آوری چندین تراکنش در یک مجموعه دستور همراه با ذخیره شماره حساب اصلی است. بنابراین در مواردی که ضعف امنیتی مربوط به ذخیره اطلاعات به وجود آید، اطلاعات مربوط به چندین تراکنش ذخیره شده در مجموعه دستورات و توافق‌های در حال انتظار می‌توانند در مدت

[۲۱] Settlement

کوتاهی فاش شوند. دزدیدن تعدادی از اطلاعات کارت در هنگام صحت‌سنجی، مستلزم شنود طولانی مدت در سیستم است.

جدول ۱-۲: عوامل موجود در صحت سنجی و فرآیند توافق

افراد شامل	مجوز	توافق
صاحب کارت	کشیدن کارت یا شماره رمز حساب در پایانه فروش	پرداخت فیش
تاجر	پذیرش، پردازش و انتقال داده صحت‌سنجی	ذخیره و انتقال داده‌های حساس صاحب کارت
سخت‌افزار تاجر (نوار مغناطس خوان، صفحه رمز)	پذیرش، پردازش و انتقال اطلاعات حساس به برنامه پرداخت	نا مشخص
نرم‌افزار تاجر	تراکنش مالی را تحلیل می‌کند؛ تراکنش مالی را به یک درگاه متناسب هدایت می‌کند	مجموعه دستور تراکنش را ذخیره می‌کند؛ اجرا و پردازش توافق در پردازشگر یا پذیرنده متناسب
پردازشگر پرداخت	ارسال درخواست به پذیرنده متناسب	ذخیره مجموعه دستورهای تراکنش اجرا و پردازش توافق در پردازشگر یا پذیرنده متناسب حساب بانکی تاجر را اعتبارمی‌بخشد
پذیرنده	ارسال درخواست به شبکه برند یا صادرکننده کارت	حساب بانکی تاجر را اعتبار می‌بخشد
برند پرداخت	درخواست را از پذیرنده تا صادرکننده انتقال می‌دهد	توافق رل تمهید می‌بخشد
صادرکننده	میزان اعتبار صاحب‌کارت را بررسی میکند پاسخ برخط را ارائه می‌دهد	به حساب بانکی صاحب کارت تراکنش ارسال می‌کند ارسال قبض به صاحب کارت

تراکنش‌های پرداخت

هر تراکنش پرداخت دو پارامتر صحت‌سنجی و تراکنش دارد. در مرحله صحت‌سنجی، پذیرنده تاجر را به‌منظور دریافت هزینه‌ای که کمتر یا برابر با مقدار صحت‌سنجی از صاحب کارت است، صحت‌سنجی می‌کند. زمانی‌که پرداخت نهایی شود، تاجر یک تراکنشی را همراه با مقدار تراکنش به‌منظور اعلان وضعیت به پذیرنده ارسال می‌کند که یادآور شود مبلغ نباید بیشتر از مقدار صحت‌سنجی باشد.

پساخرید[۲۲] در مقابل پیش‌بررسی[۲۳] یا پیش صحت‌سنجی

با توجه به نوع تراکنش و تاجر، صحت‌سنجی ممکن است برای یک مقدار پرداخت خاصی یا برای چکیده مقدار محدود انجام گیرد. برای مثال، اگر در سوپرمارکت محلی خشکبار بخرید، پایانه فروش دقیقا مقدار خریدتان را صحت‌سنجی می‌کند. این تراکنش ساده فروش پساخرید نامیده می‌شود.

اگر شما برای گاز در ایستگاه گاز پرداخت کنید، برنامه پرداخت در ابتدا یک پیش صحت‌سنجی را برای مقدار از پیش تعیین شده محدود که توسط برند کارت یا تاجران پیش‌بینی شده است انجام می‌دهد. این امر بدین دلیل است که پایانه فروش نمی‌داند که چقدر سوخت به باک باید تزریق شود. وقتی سوخت‌گیری به اتمام برسد، پایانه فروش موجود در پمپ سوخت مقدار سوخت را ارزیابی کرده و مقدار دقیق پرداختی را ارسال می‌کند. گام‌های اضافی این چنینی را پیش‌بررسی می‌گویند.

از نظر امنیتی تفاوت اساسی مابین پیش‌صحت‌سنجی و پساخرید وجود دارد که پساخرید و پیش صحت‌سنجی شامل صحت‌سنجی اطلاعات حساس است (مسیر کلی یا PAN)، در حالی که پیام اتمام فقط شامل PAN یا عدم وجود اطلاعات کارت است چون تراکنش از قبل شروع شده است، پس اتمام می‌تواند پیش صحت‌سنجی شده و شکل‌های دیگری از تایین هویت از قبیل، شماره تراکنش یا علامت‌گذاری را استفاده کند.

بی‌اعتباری[۲۴] و بازپس دادن

عادلانه است که بگوییم بی‌اعتباری و بازپس در تضاد با پساخرید یا پیش صحت‌سنجی/ پیش‌بررسی است. اگر پرداخت به اشتباه انجام گیرد، یا مشتری بخواهد کالا را بازگرداند و پولش را پس بگیرد، حسابدار، تراکنش پرداخت بازپس و یا بی‌اعتبار را انجام می‌دهد. بی‌اعتبار معمولا زمانی صورت می‌گیرد که مشتری یا تاجر می‌خواهد که کل تراکنش را لغو کند که ممکن شامل چندین روش پرداخت و گزینه‌های دیگر باشد.

---

[۲۲] Sale
[۲۳] Completion
[۲۴] Void

بازگشت یا بازپس، معمولا زمانی مورد استفاده قرار می‌گیرد که مشتری یک وسیله را بازمی‌گرداند و تاجر بجای کل مبلغ فقط باید قسمتی از مبلغ را بازگرداند.

یکی دیگر از تفاوت مهم مابین بی‌اعتباری و بازپس در این است که بی‌اعتبار بدون وصل شدن به تراکنش حراج امکان‌پذیر نیست، درحالی‌که بازپس در هر زمانی‌که بخواهد انجام می‌گیرد. تراکنش بی‌اعتبار فقط لغو کردن تراکنش موجود قبلی است، درحالی‌که بازپس، برگرداندن مبلغ به حساب صاحب کارت بدون هیچ واسطه یا ارتباطی به فعالیت قبلی است. به‌عبارتی دیگر، استفاده از بازپس برای دزدیدن پول از حساب تاجران و گذاشتن آن به حساب افراد خلاف‌کار خیلی راحت‌تر است. همچنین، تراکنش‌های بی‌اعتبار (اگر بدرستی توسط فروشندگان برنامه پرداخت یا پردازشگرها اشاره شود) لازم نیست که شامل اطلاعات حساس باشند چون تراکنش اصلی اطلاعات را حفظ کرده است.

فرآیند تعویق[۲۵]

فرآیند تعویق (یا تعویض، یا ذخیره و ادامه، یا صحت‌سنجی آفلاین) یک عمل بسیار مهمی برای ادامه تجارت تاجران است. این فرآیند، یک ویژگی را ارائه می‌دهد که توسط آن در مواقع ضروری وقتی میزبانان فرآیند پرداخت، آفلاین یا خراب شده است، بتوان فرآیند را انجام داد. اگر برنامه پرداخت به هر دلیلی نتواند صحت‌سنجی برخط را برای خریدار تحت شرایطی انجام بدهد (با توجه به نوع کارت، مقدار تراکنش، و پارامترهای دیگر)، مجاز هست که تاییدیه داخلی را انجام داده و تراکنش‌ها را برای فرآیندهای بیشتر ذخیره کند. صحت‌سنجی تعویق تقریبا می‌تواند برای حسابداران و مشتریان زودباور خیلی روش خوبی باشد.

- زمان فرآیند تراکنش برای تاییدیه آفلاین باید در مقایسه با صحت‌سنجی برخط خیلی زیاد باشد چون برنامه پرداخت می‌تواند به‌گونه‌ای برنامه ریزی شود که برای مهلت پاسخ مشخص قبل از اجازه تایید پرداخت به‌صورت داخلی منتظر بماند. مقادیر مهلت پاسخگویی توسط تکمیل‌کنندگان پرداخت بعنوان بخشی از پروتکل پیام معین می‌شود و ممکن است با توجه به وسایل ارتباطی، تفاوت‌های چشمگیری داشته باشد. هرچندکه، در بسیاری از مواقع این مقدار می‌تواند به چندین ثانیه تنظیم شود، که بطور چشمگیری بسیار زیادتر از چندین میلی‌ثانیه برای فرآیند برخط است.
فرآیند جایگزینی[۲۶] (مانند تعویق تماس) می‌تواند یکی دیگر از دلایل تاخیر زیاد در تاییدیه آفلاین باشد. بعضی از پردازشگرها، اگر ارتباط اصلی یا میزبان قطع شود نیازمند برنامه‌های کاربردی هستند تا به میزبان پشتیبان یا خطوط ارتباطی متفاوتی سوویچ کنند تا به صحت‌سنجی برخط دست بیابند.

---

[۲۵] Fallback
[۲۶] Failover

- وقتی برنامه پرداخت تاییدیه برخط را برای تراکنش پرداخت دریافت می‌کند، پاسخ میزبان شامل کد صحت‌سنجی شده است که توسط نرم‌افزار خریدار تولید شده است. این کد اغلب در زیر رسید تراکنش پرداخت قرار دارد. اگر تراکنش در حالت آفلاین انجام پذیرد، برنامه پرداخت، در طول تایید تراکنش آفلاین کد صحت‌سنجی مختص خود را تولید می‌کند. چنین کدی می‌تواند با استفاده از الگوریتم‌های مختلفی (بعضی مواقع اجرای یک شمارش یا زمان‌سنج) تولید گردد، بنابراین برای حسابدار یا مشتری خیلی راحت خواهد بود که بتواند کد ایجاد شده توسط میزبان را تشخیص بدهد. برای مثال:

  کد صحت‌سنجی برگردانده شده توسط میزبان: FVIKP0

  کد صحت‌سنجی تولید شده آفلاین توسط برنامه پرداخت: LA1234

یک ویژگی بسیار مهم فرآیند تعویق این واقعیت است که پایانه فروش باید اطلاعات حساس صاحبان کارت را در طول مدت خاموشی شبکه در یک دیسکی ذخیره کند، که ممکن است از چندین ثانیه تا چندین روز متغیر باشد. چنین نیازی به جمع‌آوری اطلاعات حساس در حجم انبوه، فرصت را به مهاجمان می‌دهد. طبیعتا، اگر سیستم به‌طور درست و کامل طراحی شده باشد، اطلاعات صاحبان کارت بعد از صحت‌سنجی، مدت زیادی در دستگاه پایانه فروش ذخیره نمی‌شود.

انقضای مهلت[۲۷]

انقضای مهلت، مکانیزمی است که از هزینه‌های تکراری جلوگیری می‌کند، که بطور مفصل در فصل ۲ مورد بحث قرار می‌گیرد. مثالی دیگر از موقعیتی که وقتی پایانه فروش باید بطور محلی عملیات ذخیره‌سازی اطلاعات صحت‌سنجی حساس را انجام بدهد TOR می‌باشد که ممکن است بعدا توسط مهاجم بازیابی شود.

انواع خاصی از تراکنش

انواع مختلف تراکنش غیر متداول وجود دارد (جدول ۱-۳) که عموما در شرایط خاص یا موقع کنترل انواع مختلف کارت صورت می‌گیرد. یک نمونه برای چنین تراکنش‌هایی استعلام و شارژ دوباره کارت هدیه است. استعلام مانده حساب برای بررسی مقدار مانده حساب کاربرد دارد، و تراکنش نهایی ممکن است شامل مسیر کلی داده باشد. شارژ دوباره به‌منظور اضافه کردن پول به کارت هدیه است که با استفاده از روش‌های پرداخت (مانند پول نقد یا کارت اعتباری) متفاوتی انجام می‌گیرد، بنابراین شامل اطلاعات حساس کارت می‌باشد.

[۲۷] Timeout Reversal

جدول ۱-۳: انواع مختلف تراکنش پرداخت

نوع تراکنش	هم ارز	عملکرد	مشکلات امنیتی
حراج	خرید	تراکنش پرداخت منظم	شامل همه اطلاعات حساس صحت‌سنجی
پیش صحت‌سنجی	مجوز	مبالغ در درسترس را بررسی کرده و مجوز را بدست می‌اورد	شامل همه اطاعات صحت‌سنجی
اتمام	تمام کردن	نهایی کردن پرداخت که توسط پیش‌صحت‌سنجی شروع شده است	ممکن است شامل PAN باشد
بی اعتبار	ارسال بی‌اعتبار	لغو تراکنش قبلی	مستلزم پیوند به تراکنش اصلی است؛ ممکن است شامل اطلاعات حساس باشد
بازپس	برگشت	حساب صاحب کارت را شارژ می‌کند	شامل همه اطاعات صحت‌سنجی
انقضای مهلت	لغو	سعی در لغو تراکنش در مواقعی که پاسخی دریافت نشود	شامل همه اطاعات صحت‌سنجی
استعلام	بررسی حساب	مانده حساب در کارت هدیه را چک می‌کند	شامل همه اطاعات صحت‌سنجی
شارژ دوباره	بارگذاری مجدد	اضافه کردن پول به حساب کارت هدیه	شامل همه اطاعات صحت‌سنجی

آسیب‌پذیری نواحی اصلی برنامه پرداخت

چندین روش برای حمله به سیستم پایانه فروش و برنامه پرداخت مشترک با آن وجود دارد که بتوان با استفاده از آن اطلاعات حساس کارت را دزدید. چنین روش‌هایی در نظریه امنیت اطلاعات اغلب مسیرهای حمله نامیده می‌شوند. مسیر حمله معمولا شامل سناریوهای حمله است—توضیح درباره گام‌ها و ابزارهای مورد استفاده است. اگر شما اطلاع دارید که یک حمله خاصی در پیش است، حداقل از لحاظ نظری، وقتی بحث درمورد کنترل امنیت برنامه باشد سناریو خاصی (برای مثال، روش‌های نفوذ، ساختار خاص، و ابزارهای مورد استفاده در حین حمله) زیاد مهم نیست (اقدامات محافظتی). بنابراین، بجای مسیرهای حمله، به نواحی آسیب‌پذیر توجه ویژه خواهد شد.

در مطالب این پژوهش هدف مورد حمله همیشه اطلاعات حساس پرداخت است (یا اطلاعات صاحبان کارت)، و محیط، برنامه پرداخت و همه اطلاعات پایانه فروش محل خرید است، ناحیه آسیب‌پذیر معمولا موقعیت (فیزیکی و منطقی) داده‌ها را در لحظه حمله در داخل برنامه معین می‌کند. سه موقعیت برای داده برای هر نرم‌افزاری که شامل برنامه پرداخت است، وجود دارد:

۱. **داده در حافظه:** وقتی برنامه پرداخت، فرآیند صحت‌سنجی یا توافق را انجام می‌دهد، دستکاری‌های مختلفی روی اطلاعات کارت در حافظه رایانه میزبان انجام می‌دهد (معمولا رم دستگاه پایانه فروش).
۲. **داده ذخیره شده:** برنامه داده پرداخت را در داخل دیسک سخت در حالت دائم یا موقت ذخیره می‌کند.
۳. **داده در حال انتقال:** برنامه پرداخت از دیگر دستگاه‌ها و برنامه‌ها اطلاعاتی را دریافت و ارسال می‌کند.

به استثنای داده در حافظه، موقعیت‌های دیگر داده، چندین زیرمجموع دارند که توسط تکنولوژی‌های مربوطه معین می‌شود. برای مثال، داده ذخیره شده می‌تواند در پایگاه داده یا فایل‌های گزارش ذخیره شوند، و داده می‌تواند با استفاده از ارتباط سریال یا LAN جابه‌جا شود.

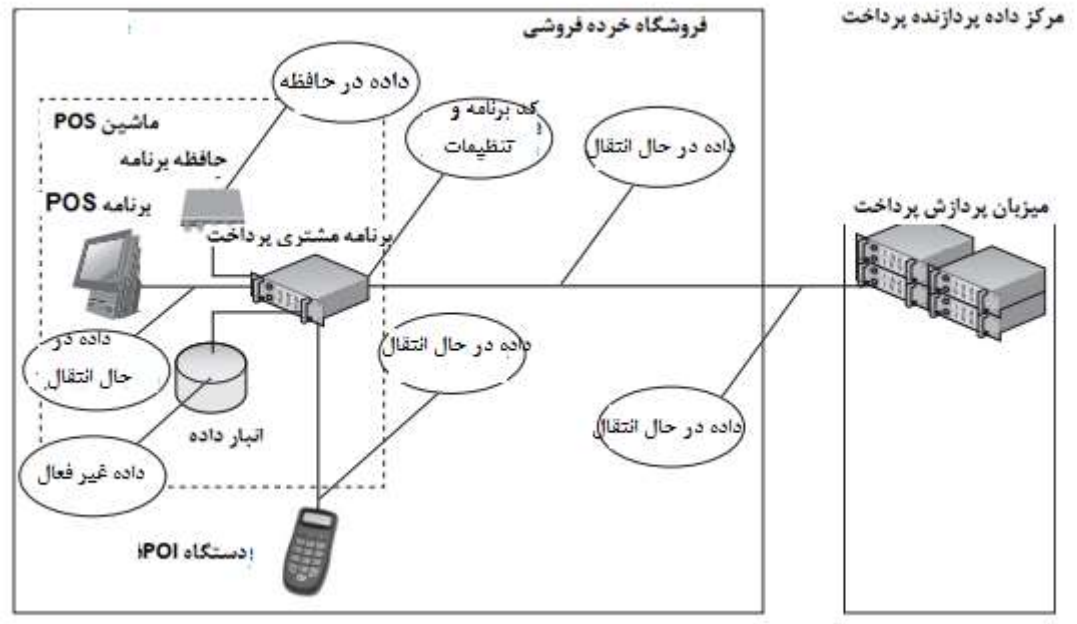
یکی دیگر از ضعف اصلی ناحیه‌ای، خود کد برنامه پرداخت و تنظیمات آن است. کد یا تنظیمات شامل هیچ نوع اطلاعاتی از صاحبان کارت نمی‌شوند، اما به‌منظور دسترسی غیرصحت‌سنجی شده به داده‌ها در نواحی آسیب‌پذیر می‌توانند با مهاجمان و نرم‌افزارهای آلوده سازگار شوند.

با این گفته، چهار ناحیه آسیب‌پذیر برای برنامه‌های پرداخت وجود دارد که در شکل ۱-۵ نشان داده شده است:

۱. داده در حافظه

۲. داده ذخیره شده یا غیرفعال
۳. داده در حال انتقال
۴. کد برنامه و تنظیمات

جدول ۱-۴ آسیب‌پذیری نواحی را با زیرنواحی لیست کرده است.



شکل ۱-۵: نواحی آسیب‌پذیر اصلی

جدول ۱-۴: نواحی آسیب‌پذیر برنامه‌های پرداخت

ناحیه کلیدی	زیرناحیه	مثال‌ها	داده معمول	محافظت شده؟
داده در حافظه			کامل	خیر
داده غیرفعال	ذخیره موقت	S&F, TOR	کامل	بله
	ذخیره بلند مدت	مجموعه، توافق، بایگانی	PAN	بله
	فایل‌های ثبت شده		تصادفی	
داده در حال عبور	ارتباطات محلی	LAN مابین ماژول‌های برنامه	کامل	خیر
	ارتباطات مابین نقطه تبادل و پایانه فروش		کامل	خیر

	ارتباطات پردازشگرها	پیوند میزبان	کامل	خیر
کد و تنظیمات برنامه	کد برنامه		نامشخص	خیر
	تنظیمات برنامه		نامشخص	خیر

خلاصه فصل

انواع مختلف کارت پرداخت اعتباری، بانکی، هدیه و ناوگانی می‌باشند. کارت اعتباری و بانکی از همه بیشتر آسیب‌پذیر هستند چون مورد مقبولیت اکثر افراد بوده و در حجم وسیعی از داد و ستدها استفاده می‌شوند.

چندین عامل اصلی در فرآیند پرداخت الکترونیکی وجود دارد که از جمله آنها می‌توان به صاحبان کارت، تاجر، فروشندگان نرم‌افزار، تولیدکنندگان سخت‌افزار، درگاه، پردازشگرها، پذیرندگان، برند کارت و صادرکنندگان اشاره کرد. تاجران از جمله فروشگاه‌ها و پایانه فروش‌ها آسیب‌پذیرترین عامل در این زنجیره هستند چون بطور مستقیم با عموم سروکار دارند، و تبادل آن با مشتریان سطوح بسیار گسترده‌ای را دارد.

فرآیند پرداخت توسط کارت شامل دو مرحله اصلی است: صحت‌سنجی و توافق. فاز صحت‌سنجی خیلی خطرناک است، چون مستلزم انتقال اطلاعات حساس صحت‌سنجی است که ممکن است در بعضی مواقع این اطلاعات در طول چندین سیستم رمزنگاری نشده باشد. چنین داده‌هایی می‌تواند توسط مهاجم مورد سوء استفاده قرار گرفته و برای تولید کارت‌های تقلبی استفاده شوند.

چندین ناحیه مهم آسیب‌پذیر برای سیستم پایانه فروش برنامه پرداخت مشترک با آن وجود دارد:

- داده در حافظه
- داده ذخیره شده یا غیرفعال
- داده در حال انتقال
- تنظیمات و کد برنامه

هریک از نواحی آسیب‌پذیر ویژگی‌های خود را دارد و با بکارگیری روش‌های مختلف و در زمان‌های متفاوت در طول چرخه پرداخت می‌تواند مورد حمله قرار گیرد.

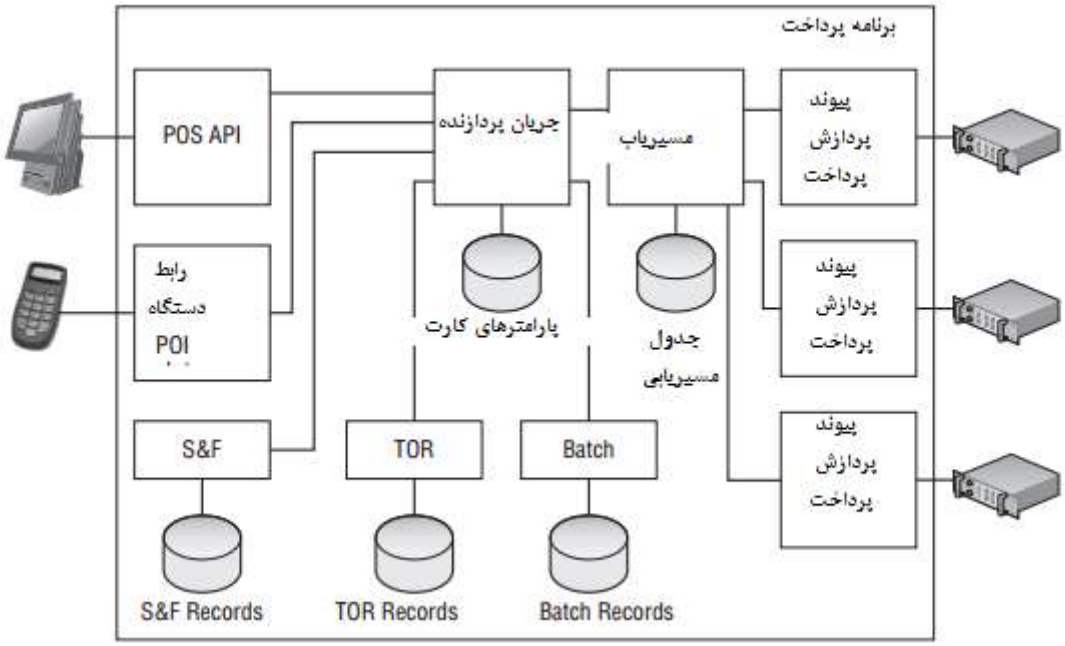
فصل دوم:

معماری برنامه پرداخت

به منظور درک کامل انواع مختلف تهدیدات که ممکن است برنامه پرداخت را مورد حمله قرار بدهد، ابتدا ضروری است که در مورد ساختار اینترنتی این سیستم‌ها بحث شود. جزئیات پیاده‌سازی هر فروشنده با فروشنده دیگر ممکن است متفاوت باشد، اما در طراحی، ویژگی‌های اصلی مشابه یکدیگرند.

بخش‌های الزامی برنامه پرداخت

معماری پرداخت معمولی در شکل ۲-۱ نشان داده شده است که شامل ماژول‌های فرآیند و رابط است. رابط پلی به دنیای بیرون است. ماژول‌های در حال پردازش تراکنش پرداخت را کنترل می‌کنند.



شکل ۲-۱: بلوک‌های معماری یک برنامه پرداخت معمولی

رابط‌ها

همه سیستم‌ها مستلزم ارتباط با دنیای بیرون مانند سخت افزار جانبی و نرم‌افزارهای خارجی هستند، بنابراین دستگاه و رابط‌های کاربری برنامه بخش مهمی از برنامه پرداخت به شمار می‌روند. سه نوع رابط کاربری خارجی وجود دارد که به برنامه پرداخت از طریق دستگاه‌ها و برنامه‌ها متصل است:

۱. رابط کاربری دستگاه POI
۲. POS API
۳. پیوند پردازشگر پرداخت

یک برنامه پرداخت ممکن است چندین رابط کاربری پیاده‌سازی شده داشته باشد، که وابسته به تعداد لوازم جانبی پشتیبانی شده مانند مدل‌های پایانه فروش و شبکه‌های صحت‌سنجی می‌باشد.

رابط دستگاه POI

رابط‌کاربری دستگاه POI وظیفه تبادل داده با صفحه‌رمز یا دستگاه‌های مستقل MSR را برعهده دارد. این رابط شامل پیاده سازی ارتباط خاص دستگاه و پروتکل ارسال و دریافت پیام است (یا چندین پروتکل دیگر، اگر دستگاه انواع مختلفی از لوازم جانبی را پشتیبانی می‌کند). ارتباطات معمولی با پورت‌های COM یا از طریق TCP/IP بر روی LAN انجام می‌شود. با این حال، پروتکل‌های ارسال و دریافت پیام متفاوت هستند، و فروشندگان زبان‌های مختص خودشان را پیاده‌سازی می‌کنند. نمونه‌های اخیر اجازه استفاده از روش‌های رمزگذاری شده از قبیلSSL یا رمزنگاری محموله داده و همچنین صحت‌سنجی همراه با تاییدیه برای امنیت اضافه را می‌دهد.

ویژگی مشترک مابین بسیاری از دستگاه‌های POI ارتباطی و پروتکل‌های ارسال و دریافت پیام، عدم وجود مکانیزم‌های امنیتی اضافی در داخل ارتباطات برای محافظت از داده‌های صاحبان کارت است. پیاده سازی-های پیش‌فرض فرآیند رمزنگاری داده‌های حساس و صحت‌سنجی دستگاه توسط برنامه پرداخت را برآورده نمی‌کند و بدین معناست که داده‌های حساس به‌راحتی می‌توانند شنود شده یا دستگاه‌ها می‌توانند جایگزین شوند.

API پایانه فروش

رابط کاربری پایانه فروش مسئول ارتباط با برنامه پایانه فروش و به منظور کنترل جریان تراکنش پرداخت است، که شامل دریافت پارامترهای تراکنش، پردازش حسابدار و دستگاه‌های مشتری، و برگشت دادن نتایج می‌باشد. این نوع ارتباط از روش‌های متفاوت دیگری می‌تواند انجام بگیرد که بسته به طراحی خاص برنامه براساس پایانه فروش و برنامه پرداخت ، از تبادل داده‌های در حال پردازش حافظه تا TCP/IP از راه دور یا اتصالات HTTP طراحی شده است. عموما به آن API گفته می‌شود چون یک برنامه پرداخت چندین برنامه پایانه فروش را پشتیبانی می‌کند، و این افزونه توسط توسعه دهندگان پایانه فروش با استفاده از ویژگی‌های API ارائه شده از طریق فروشنده برنامه پرداخت صورت می‌گیرد.

نگرانی‌های امنیتی از اینکه مکانیزم‌های امنیتی استانداردی موجود نیست همانند دستگاه رابط است. مشکلات خاصی بسته به نوع برقراری ارتباط وجود دارد. اگر پایانه فروش و برنامه پرداخت تحت یک پردازه سیستم عامل اجرا بشوند، حافظه پردازش را می‌توان با استفاده از برنامه‌های جمع‌آوری حافظه بمنظور بازیابی اطلاعات حساس، اسکن کرد. در مواقعی که ارتباط از راه دور به رابط با استفاده از پروتکل‌هایی از قبیل TCP/IP ممکن باشد، آسیب‌پذیری‌ها، مشابه هر شبکه ارتباطی دیگر خواهند بود.

فروشندگان برنامه پرداخت ممکن است اذعان داشته باشند که ارتباطات مابین پایانه فروش و برنامه پرداخت اطلاعات مهمی را شامل نمی‌شود، چون برنامه پرداخت تمام جوانب هر تراکنش پرداخت را در نظر می‌گیرد و در آخر نتایج پوشیده شده را حتی بدون افشای جزئیات نوار مغناطیسی به پایانه فروش برمی‌گرداند. در چنین بیانیه‌هایی فقط قسمتی از آن برای نگرانی‌های پیش‌رو حقیقت دارد. اولا، در هنگام ورود دستی، وقتی MSR در خواندن نوار مغناطیسی با شکست مواجه می‌شود، PAN معمولا توسط حسابدار در پایانه فروش وارد می‌شود و به برنامه پرداخت همراه با داده حساس مانند تاریخ انقضا، کد CVV یا ZIP ارسال می‌شود. دوما، رابط برنامه پرداخت ممکن است که روش‌های بازگشت PAN بدون پوشش را یا حتی تمام مسیرها را در یک فایل متنی افشا کند. چنین روش‌های API اگر توسط کنترل دسترسی محافظت نشود، می‌تواند توسط نرم‌افزارهای مخرب دستکاری شود.

پیوند پردازشگر پرداخت

پیوند پردازشگر دو وظیفه عمده را انجام می‌دهد:

۱. پارامترهای تراکنش را از برنامه داخلی به یک فرمت خاصی که بر اساس پروتکل پیام پردازنده پرداخت است تبدیل می‌کند.
۲. با استفاده از پروتکل ارتباطات پشتیبانی شده توسط پردازشگر پرداخت، با میزبان معتبر ارتباط برقرار می‌کند.

هر پیوند پردازشگر بصورت کد-ثابت است تا با یک پردازشگر خاصی ارتباط برقرار کند. هر چند، موقعیت سرور پردازشگر معمولا کد-قابل‌تغییر است. برای مثال، تنظیمات میزبان ممکن است شامل آدرس IP و سرور با پروتکل TCP/IP در یک شبکه خصوصی باشد، یا آدرسی برای پروتکل HTTP در اینترنت باشد.

اگر تنظیمات محافظت نشود و هیچ صحت‌سنجی مابین مشتری و سرور نباشد، پارامترهای گفته شده می‌تواند با برنامه پرداخت در تداخل باشد که ممکن است در نهایت منجر به ارتباط با میزبان جعلی بشود. چنین تطبیقی ممکن است از دید سیستم پرداخت مخفی باشد چون میزبان جعلی ممکن است همه ترافیک را شنود کرده (بدلیل به سرقت بردن اطلاعات مهم صاحبان کارت) و این ترافیک را برای یافتن پردازنده اصلی و قانونی ردیابی کند (پس نفوذ به سرور مخفی خواهد ماند).

پیوندهای خاصی برای توسعه آزمایشی یا برنامه داخلی ممکن است ایجاد شود. چنین پیوندهای ساختگی لزومی ندارد که برای صحت‌سنجی با محیط بیرون ارتباط برقرار کند، اما براحتی می‌تواند به گونه‌ای برنامه‌ریزی شود که بتواند بطور خودکار پاسخ‌های تایید که براساس کمترین یا بدون شرط است را برگشت بدهد. اگر تراکنش‌های کارت عادی به چنین پیوندی هدایت شود، پرداخت توسط پایانه فروش بدون ثبت هیچ تراکنش واقعی انجام خواهد شد.

ماژول‌های فرآیند

دومین گروه از بخش‌های برنامه پرداخت شامل ماژول‌های پردازش در حال جریان است که فرآیند پرداخت را از لحظه کشیدن کارت تا پرداخت به تاجر و شارژ حساب صاحب کارت را پردازش می‌کند. ماژول‌های پردازش اصلی شامل موارد زیر می‌شود:

- مسیریاب
- ذخیره و ادامه
- مهلت انقضا
- مجموعه دستور

مسیریاب

ماژول مسیریاب وظیفه ارسال مسیر تراکنش را برای صحت‌سنجی، تکمیل یا توافق به یک پیوند پردازشگر پرداخت برعهده دارد که این پیوند بر اساس پارامترهای از پیش تعیین شده محدوده BIN کارت، نوع کارت و نوع تراکنش است. معمولا، برنامه پرداخت بیش از یک پیوند پردازشگر پرداخت دارد که در سطوح مختلفی پیاده‌سازی شده است و پایانه فروش، ذخیره، یا تغییر مرکز داده وابسته به معماری آن سیستم مورد نظر است. برای نمونه، پرداخت‌های کارت اعتباری می‌تواند به پردازشگر استاندارد هدایت شود؛ تراکنش‌های PIN می‌تواند به یک شبکه بانکی خاصی ارسال شود؛ و کارت‌های هدیه می‌تواند بطور اختصاصی توسط سوییچ حلقه بسته شده ارسال شود. به‌منظور اتخاذ تصمیم، مسیریاب از جدول مسیریابی استفاده می‌کند که حداقل شامل مقادیر PAN با اشاره‌گرهایی به پیوندهای پردازش پرداخت خاصی است. برای مثال، طبق تنظیمات تعریف شده در جدول ۲-۱، همه تراکنش‌های انجام شده با کارت ویزا برای صحت‌سنجی باید به بانک آمریکا ارسال شود، در حالی‌که تراکنش‌های کارت American Express به طور مستقیم به بانک مرجع هدایت می‌شوند.

جدول ۲-۱: نمونه‌هایی از محدوده مسیریابی PAN

PAN از	PAN تا	شناسه پیوند
4000000000000000	4999999999999999	BOA
340000000000000	349999999999999	AMEX
3700000000000000	379999999999999	AMEX

تصمیمات مسیریابی براساس عواملی از قبیل: نوع تراکنش اتخاذ می‌شود. برای مثال، همه فعالسازی، بارگذاری مجدد، و استعلام مبلغ در کارت‌های هدیه می‌تواند به‌نحوی به شبکه پردازش کارت هدیه هدایت شود که به PAN لزومی نداشته باشد.

یک شخص آشنا به طراحی برنامه پرداخت (مانند کارمند سابق) ممکن است که تنظیمات مسیریابی را به گونه‌ای دستکاری کرده (اگر در مقابل خرابکاری محافظت نشده باشد) و سیستم را مجبور کند که تراکنش‌ها را با کارت‌های جعلی بدست آمده از PAN پیوندهای پردازشگر ساختگی ادامه بدهد. چنین تراکنش‌هایی تاییدیه قانونی را بدون ثبت پرداخت واقعی، از پایگاه‌داده پردازنده دریافت خواهد کرد.

ذخیره و ارسال یا S&F

ماژول ذخیره و ارسال یک عمل تعویق بسیار مهمی را پیاده‌سازی می‌کند که به تاجران این اجازه را می‌دهد که تجارت خود را با پذیرش بدون وقفه پرداخت‌های الکترونیکی ادامه بدهد.

وقتی شبکه صحت‌سنجی خراب شود و تراکنش توسط پردازشگر پرداخت، به صورت آفلاین فرآیند را انجام خواهد داد، ماژول ذخیره و ارسال تاییدیه آفلاین را ایجاد کرده و تراکنش را در پایگاه داده ذخیره و ارسال، ذخیره می‌کند. به محض این‌که شبکه صحت‌سنجی دوباره درست شود، ذخیره و ارسال تراکنش‌های ذخیره شده را به میزبان ارسال می‌کند. راه‌حل دیگر ارسال تراکنش های ذخیره شده بصورت پردازش آخروقت به‌منظور توافق است.

از نظر امنیتی، مهم‌ترین ویژگی ذخیره و ارسال این است که، سوابق تراکنش‌های آفلاین ممکن است در دستگاه پایانه فروش یا BO برای مدتی گیر کند که این مدت با توجه به محدوده شبکه ممکن است چند ثانیه تا چند روز متغیر باشد. اغلب چنین سوابقی شامل کل مسیر داده‌های ضروری برای پردازش میزبان است. استانداردهای PCI DSS و PA-DSS اجازه ذخیره کل مسیر را نمی‌دهد.

انقضای مهلت یا TOR

انقضای مهلت یک مکانیزم کنترل خطا است که از هزینه (خرید) تکراری در حساب صاحب‌کارت جلوگیری می‌کند. خریدهای تکراری زمانی رخ می‌دهد که پایانه فروش، پاسخ معتبری از میزبان هنگامی‌که تراکنش توسط پردازشگر تایید و ثبت شده است، دریافت نکند (مهلت پاسخ نامیده می‌شود). در این موقع، تراکنش برای نهایی کردن پرداخت در پایانه فروش، حسابدار، و دستگاه مشتری ظاهر نمی‌شود اما از حساب مشتری کم شده است. اگر حسابدار کارت را بکشد و تاییدیه را دریافت کند، از حساب مشتری دوبار کم می‌شود. برای مقابله با چنین شرایطی، وقتی برنامه پرداخت از طرف میزبان پاسخی دریافت نمی‌کند (یا به‌عبارتی پیام مدت پاسخ دریافت می‌کند) این برنامه پیام انقضای مهلت را ایجاده کرده و ارسال می‌کند و میزبان را حتی بدون توجه به تایید شدن یا نشدن آن مجاب به لغو تراکنش می‌کند. اگر ماژول انقضای مهلت با

میزبان ارتباط برقرار نکند، پیام‌های انقضای مهلت را در پایگاه داده خودش جمع می‌کند و به محض برخط شدن شبکه به سرور ارسال می‌کند. این فرآیند خیلی شبیه به مکانیزم ذخیره و ارسال است. چالش امنیتی اصلی برای این ماژول مشابه ماژول ذخیره و ارسال یعنی ذخیره داده‌های صحت‌سنجی حساس است.

با استفاده از ویژگی‌های انقضای مهلت و ذخیره و ارسال، افشای داده‌های حساس صحت‌سنجی به رسانه‌های جدید گسترش یافته و برای بازه زمانی طولانی‌تری ماندگار شده است (بجای چند میلی ثانیه به چندین دقیقه تا چندین روز)

مجموعه دستور[۲۸]

ماژول مجموعه دستور (با اسامی، مجموعه دستور خاتمه، تغییر نزدیک، اتمام روز یا ماژول توافق شناخته می‌شود) مسئول ثبت و توافق تراکنش‌های پرداخت پردازش شده در حین یک بازه زمانی خاص، معمولا یک روز کاری است.

کلمه Batch برگرفته از روزهایی است که کارت‌های اعتباری بصورت دستی و با استفاده از دستگاه مخصوص حکاکی دستی پردازش و حک می‌شدند. منظور از حک، مسیرهایی از PAN، تاریخ انقضا و نام صاحب‌کارت را بر روی قبض تراکنش چاپ برجسته کردن است (بر روی کارت بانکی برجسته شده است). در حین روز کاری، ورق‌های کاغذ در batch ذخیره شده و به شرکت‌های اعتباری به منظور توافق ارسال می‌گردد. امروزه، فرآیند توافق بطور کلی مشابه گذشته است تنها با یک تفاوت که این فرآیند توسط رایانه صورت می‌گیرد.

چندین نکته مهم درباره فرآیند خاتمه داده مجموعه دستور مستلزم توجه بسیاری است. مسئله اول، بعضی از پردازنده‌ها هنوز هم مستلزم ارسال PAN کامل در طول توافق هستند، بدین معنی که تعداد زیادی شماره حساب توسط برنامه پرداخت در طول روز جمع شده است که بعدا برای توافق ارسال خواهد شد. همانطور که می‌توانید تصور کنید، این حقیقت توسط افراد بدون توجه نخواهد ماند. هرچندکه، بعضی از پردازشگرهای پرداخت، رابط‌های خودشان را تغییر داده‌اند که نیازی به ارسال PAN برای توافق نخواهد بود.

مسئله دوم کمی جدی‌تر است چون هنوز راه‌حلی وجود ندارد. این کار باید از طریق مصالحه و بازپرداخت صورت گیرد. اگر مبلغ کل پردازش شده پرداخت در میزبان با مبلغ کل محاسبه شده توسط سیستم ذخیره پرداخت مطابقت نداشته باشد بعضی از پروتکل‌ها مستلزم ارسال دوباره سوابق تراکنش در آخر روز می‌شوند. این فرآیند اصلاح خطا مصالحه نامیده می‌شود. در حین توافق آخر روز و مصالحه، تراکنش‌ها توسط پردازشگر می‌تواند برای هر دلیلی رد شود. برای مثال، پرداخت ممکن است بصورت آفلاین تایید شود و توسط ماژول ذخیره و ارسال پردازش شود. چنین عدم پذیرشی بازپرداخت نامیده می‌شود. مشکل مصالحه و بازپرداخت این است که مستلزم ارسال دوباره و کامل PAN به میزبان هستند، و بدین معنی است که همه

[۲۸] Batch

شماره حساب‌ها باید حداقل چندین روز بایگانی شوند. این ویژگی ماژول فرآیند مجموعه دستور برای بعضی از افراد جالب است.

ذخیره داده

علاوه بر داده در حال عبور که بیشتر با رابط‌ها و پروتکل‌ها در حال ارتباط هستند، موقعیتی دیگر به‌نام داده ذخیره شده وجود دارد، این اصطلاح بکار گرفته شده است تا هر شکلی از ذخیره سازی هارد دیسک از قبیل، فایل داده ساده یا فایل گزارش را شامل شود. چندین مورد کاربردی برای ذخیره داده‌ها در برنامه‌های پرداخت وجود دارد. انقضای مهلت، ذخیره و ارسال، سوابق مجموعه دستور و برنامه ثبت فایل‌ها از جمله انها می‌باشند. حتی برنامه‌های منحصرا پرداخت هم می‌تواند تکنولوژی‌های (و مکانیزم‌های حفاظتی) مختلفی را برای ماژول‌های مختلفی به‌کار گیرد. برای مثال، هنگامی که گزارش‌های برنامه بعنوان یک فایل ساده می‌تواند ذخیره شود، انقضای مهلت، ذخیره و ارسال و مجموعه دستور می‌تواند از پایگاه داده‌ای مانند از MS SQL Server استفاده کند. در این موقع، برنامه پرداخت ممکن است برای فایل‌های تخت و پایگاه داده‌ها روندهای مختلفی از رمزنگاری را پیاده سازی کند.

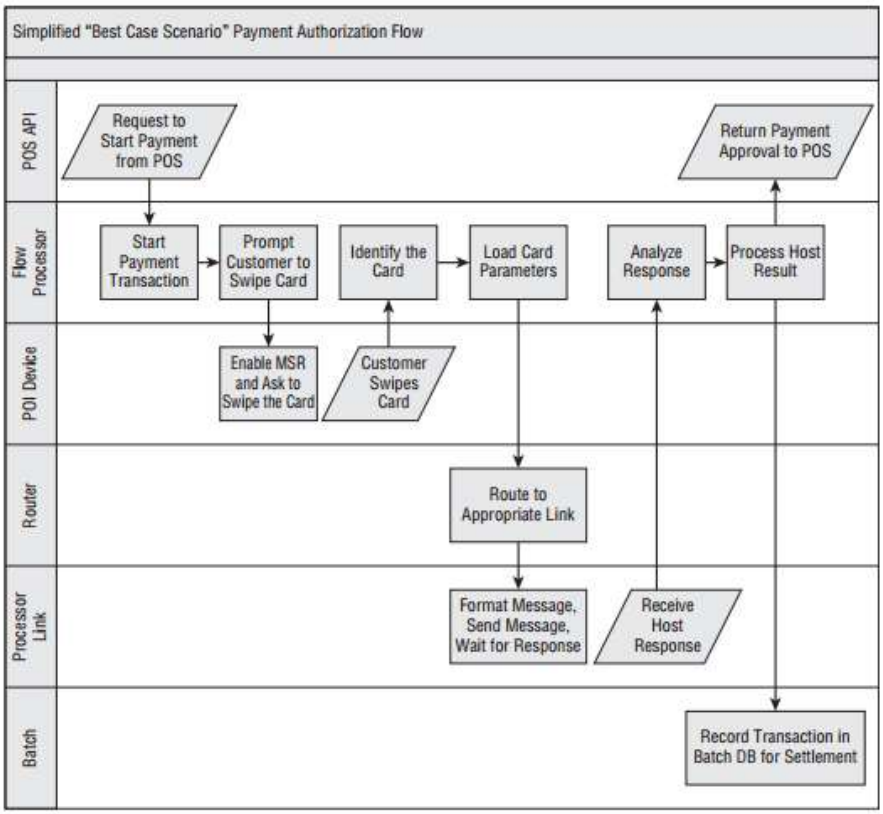
برخلاف پروتکل‌های ارتباط و ارسال و دریافت پیام، هیچ تلاشی برای تاسیس استانداردهایی در مورد ذخیره داده‌های برنامه پرداخت انجام نشده است. چندین تکنولوژی در دسترس وجود دارد، و فروشندگان نرم‌افزار در انتخاب مشتریان مورد علاقه خود آزاد هستند. در بسیاری از مواقع، هیچ استاندارد خاصی برای رمزنگاری داده‌های ذخیره شده وجود ندارد. بنابراین توسعه‌دهندگان، یک رمزنگاری اختصاصی را به‌وجود آورده و کلید مربوطه نیز از یک الگوریتم بسیار قوی که توسط استانداردهای شرکت مورد نظر است پیروی می‌کند. در عمل، بدین معنی است که برنامه استانداردهای کتابخانه رمزنگاری را فراخوانی می‌کند. هرچندکه، پیاده‌سازی کلی مکانیزم رمزنگاری در این برنامه‌ها هنوز هم اختصاصی است و بدین معنا است که سطح آسیب‌پذیری آن نامشخص است.

کد هر الگوریتم رمزنگاری تنها یک طرف مکانیزم رمزنگاری را شامل می‌شود و حتی قویترین الگوریتم‌های رمزنگاری هم اگر الگوریتم و مدیریت کلید ضعیف داشته باشند با شکست مواجه می‌شوند. DUKPT برای محافظت از PIN بانکی یک استثنا است و یک نمونه از تکنولوژی استاندارد شده است که نه تنها الگوریتم رمزنگاری را تعریف می‌کند بلکه مدیریت کلید و حتی محیط فیزیکی اطراف آن را هم تعریف می‌کند. استفاده از چندین تکنولوژی ذخیره داده ممکن است مستلزم استفاده از مکانیزم مختلف محافظتی رمزنگاری باشد، که آسیب‌پذیری برنامه را افزایش می‌دهد. عدم وجود تکنولوژی استاندارد امنیتی برای ذخیره داده‌های برنامه پرداخت موجب ایجاد چندین ضعف مرتبط با داده ذخیره شده می‌شود.

چرخه معمول تراکنش پرداخت

به‌منظور درک چگونگی ارتباط ماژول‌های یاد شده با یکدیگر، یک سناریوی تراکنش پرداخت را با هم مرور کنیم و مشاهده می‌کنیم که چگونه داده‌های صحت‌سنجی حساس مابین ماژول‌های برنامه پرداخت، در حال گردش است.

شکل ۲-۲ نشان می‌دهد که ساده‌ترین چرخه هم به سه زیر چرخه وابسته به هم تبدیل شده و کل فرآیند رویداد محور است.



شکل ۲-۲: چرخه تراکنش پرداخت معمولی

از نظر برنامه پرداخت، همه چیز زمانی شروع می‌شود که حسابدار بررسی لوازم را تمام کرده و دکمه پرداخت را فشار می‌دهد (مدل‌های مختلف پایانه فروش با نام‌های مختلفی آن‌را نامگذاری می‌کنند ولی ایده یکسان است)، که کنترل را از پایانه فروش به برنامه پرداخت انتقال می‌دهد. مشتری باید کارت را بکشد. به محض کشیدن کارت، کارت شناسایی شده و پیام درخواست صحت‌سنجی ساخته شده و به پیوند پردازنده پرداخت هدایت می‌شود. (براساس نوع کارت و پارامترهای دیگر که از پایگاه داده تنظیمات بارگذاری شده است). پیوند پردازش، پیام را طبق قوانین پروتکل شکل می‌دهد، با میزبان یک اتصال ایجاد کرده و با استفاده از پروتکل ارتباطی یک پیامی را به‌منظور صحت‌سنجی ارسال می‌کند. در این سناریو، همه‌چیز

خیلی خوب است، بنابراین برنامه به‌سرعت پاسخ تاییدیه را از میزبان دریافت می‌کند، و نتایج در پایگاه داده مجموعه ثبت می‌شود و به پایانه فروش اطلاع می‌دهد که پرداخت انجام شده است.

همانگونه که در شکل ۲-۲ مشاهده می‌کنید، حتی در ساده‌ترین موقعیت (با تاییدیه برخط مستقیم، بدون استثنا، منوی مشتریان، ذخیره و ارسال، انقضای مهلت و قبل توافق) اطلاعات صاحب‌کارت در همه مراحل فرآیند پرداخت موجود است. در یک تراکنش مادام‌العمر، داده‌های حساس در حین همه موقعیت‌های ممکن در حافظه، در حال ذخیره شده و در حال انتقال جمع می‌شود و برای حمله، احتمالات بی‌پایان وجود دارد.

ارتباط مابین ماژول‌ها

علاوه بر API خارجی و دستگاه رابط‌ها، ماژول‌های برنامه پرداخت مختلف با استفاده از پروتکل پیام داخلی باید با همدیگر ارتباط داشته باشند که این پروتکل یک فرمت اختصاصی را برای دستورات و پیام‌ها تعریف می‌کند. در نگاه اول این ارتباط مهم نیست، به‌ویژه اگر ماژول‌ها به یک باینری تبدیل شود یا در همان دستگاه جاسازی شود.

چندین خطر از نوع شنود و خرابکاری وجود دارد اگر:

- ماژول‌ها در رایانه‌های مختلف قرار بگیرد، یا
- در همان دستگاه قرار دارند اما توسط عملکردشان می‌توان متمایز کرد و API مخصوص خود را دارد

بخش پیش‌رو جزئیات بیشتری را راجع به ارتباطات برنامه پرداخت داخلی ارائه خواهد داد.

اتصال‌های فیزیکی

ارتباط اولیه پایانه‌های پرداخت کارت‌های اعتباری با شبکه‌های پرداخت از طریق خطوط تلفن و با استفاده از مودم صورت می‌گیرد. مودم دستگاهی است که سینگال‌های سریال ارتباطی از کامپیوتر را به موج الکتریکی ترجمه می‌کند که می‌تواند توسط سیم تلفن معمولی جابه‌جا شود. این ارتباط خیلی آهسته می‌باشد چون برای یک مودم ساده، ایجاد یک اتصال به سرور زمان زیادی را می‌خواهد. هرچندکه این روش مزایایی را نیز دارا می‌باشد، یکی از آن‌ها این بود که اطلاعات حساس را نمی‌توان توسط کنترل از راه دور بدست آورد.

بعضی از تاجران هنوز از مودم‌های dial-up استفاده می‌کنند، درحالی‌که برخی آن‌ها را بعنوان مکانیزم ضد شکست برای شرایطی که ارتباط شبکه اصلی در Leased line، frame relay یا اینترنت به‌هر دلیلی قطع باشد. در این مواقع، برنامه پرداخت به‌صورت خودکار به خط جایگزین dial-up تغییر یافته، و فرآیند پرداخت بدون وقفه صورت می‌گیرد. به محض برخط شدن اتصال شبکه، برنامه پرداخت به کانال اصلی خود بازمی‌گردد.

سیستم‌های Leased line و frame relay توسط تاجران بعنوان ارتباط اساسی برای ارتباطات شبکه سنتی مورد استفاده قرار می‌گیرد. شکل‌های ارتباطی مشترکی مابین فروشگاه‌های خرده فروش و مرکز نظارت یا پردازشگرهای پرداخت وجود دارد. مزیت امنیتی آشکار WAN (خودش را از شبکه‌های عمومی مانند اینترنت ایزوله می‌کند) در شرایط مشکل‌ساز این است‌که استانداردهای امنیتی PCI مستلزم رمزنگاری برروی این شبکه‌ها نیستند. ارتباطات سریال در RS232 به‌طور اختصاصی مابین دستگاه POI و رایانه میزبانی کننده پایانه فروش و برنامه‌های کاربردی استفاده می‌شود. مودم‌های dial-up از پورت‌های سریال COM برای ارتباط با رایانه میزبانشان استفاده می‌کند. هرچندکه، پورت سریال نسبت به شنود شبکه سنتی آسیب‌پذیر نیست، ولی می‌تواند توسط شنودکنندگان سریال مورد شنود قرار بگیرد.

پروتکل‌های ارتباطی

تبادل پیام مابین ارسال کننده و دریافت کننده توسط پروتکل‌های ارتباطی صورت می‌گیرد که در سطوح بالایی پشته اتصال سیستم باز قرار دارد (در جدول ۲-۲ نشان داده شده است). ماژول برنامه پرداخت، این پروتکل ارتباطی را پیاده سازی می‌کند که مسئول ایجاد ارتباط، تحویل بیت‌های پیام از ارسال کننده به دریافت‌کننده، و کنترل خطا است. پروتکل‌های ارتباطی سطح بالا می‌توانند چندین سطح از پشته اتصال سیستم باز را استفاده کنند، وقتی‌که این پروتکل با برنامه لایه ۷ پشته اتصال سیستم باز ارتباط می‌یابد معمولا برای برنامه پرداخت آشکارا خواهد بود.

سرویس‌های وب یک نمونه از پروتکل پیچیده در ارتباط با چالش‌های امنیتی بر روی سطوح مختلف پشته اتصال سیستم باز است.

جدول ۲-۲: ارتباطات برنامه پرداخت و پشته اتصال سیستم باز

شماره لایه	نام لایه	ارتباطات برنامه پرداخت	مثال
لایه ۷	برنامه	پروتکل‌های ارتباطات	HTTP, SOAP
لایه ۶	ارائه	پروتکل‌های ارتباطات	SSL
لایه ۵	نشست	پروتکل‌های ارتباطات	Name Pipes, RPC, Full Duplex
لایه ۴	انتقال	پروتکل‌های ارتباطات	TCP
لایه ۳	شبکه	پروتکل‌های ارتباطات	IP
لایه ۲	پیوند داده	اتصالات فیزیکی	Ethernet, Frame Relay (WAN)

لایه ۱	فیزیکی	اتصالات فیزیکی	DSLR, RS-232, 10BASE

ارتباط محلی

رابط‌های محلی و APIهای پایانه فروش از پروتکل ارتباطی مانند DLL API یا Windows COM استفاده می‌کنند. اگر مشتری (برنامه پایانه فروش) و سرور (EPS) تحت فرآیند یکسان سیستم عامل (در فضای آدرس یکسان) اجرا شوند، فراخوانی‌های DLL APIs و COM در حال پردازش مورد استفاده قرار می‌گیرد. COM خارج از پردازش می‌تواند برای ارتباطات مابین فرآیندهای مختلف مورد استفاده قرار گیرد وقتی‌که پایانه فروش و EPS (یا دو ماژول برنامه پرداخت داخلی) برنامه‌های جداگانه قابل اجرا باشند.

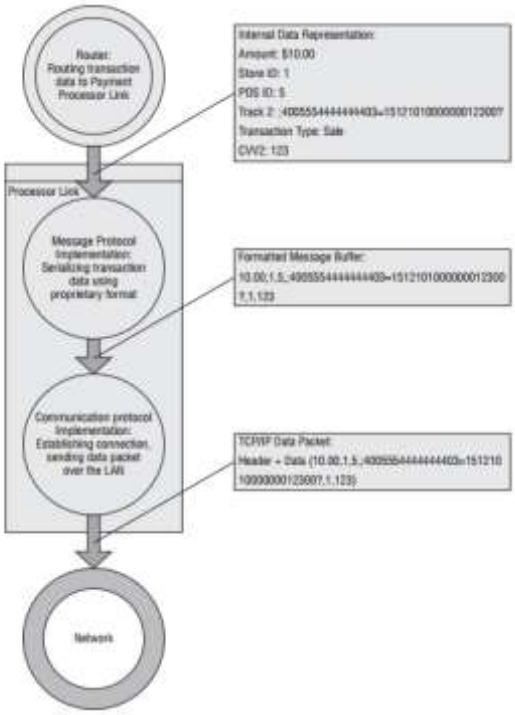
ارتباطات در حافظه محلی به شنود کننده راه دور افشا نمی‌شود چون هیچ شبکه ارتباطی درگیر وجود ندارد. هرچندکه، چنین ارتباطی در حافظه رم افشا می‌شود.

پروتکل‌های ارسال و دریافت پیام

وقتی صحبت از ارتباط برنامه پرداخت می‌شود، لازم است که تفاوت مابین دو پروتکل ارسال و دریافت پیام و ارتباط مشخص گردد که از نظر امنیتی خیلی مهم است.

پروتکل‌های ارسال و دریافت پیام در سطح نرم‌افزار برنامه در بالای پشته اتصال سیستم باز کار می‌کند. پروتکل های پیام دو امر مهم را انجام می‌دهند:

۱. گفت‌وگو (با استفاده از یک فرمت پیام مخصوص) بر اساس پارامترهای تراکنش (مانند مقادیر دلار، مسیر ۲، و PAN) از ارائه دهنده برنامه داخلی آن‌ها تا به شکل غیر قابل فهم ارسال شود و توسط جفت دیگر کانال ارتباطی دریافت شود. این فرآیند مکالمه معمولا سریال‌سازی پیام نامیده می‌شود. وقتی پیام تراکنش توسط ارسال کننده سریال سازی شد، این پیام با استفاده از پروتکل ارتباط می‌تواند به دریافت کننده ارسال شود، دریافت‌کننده‌ای که عکس این کار را انجام می‌دهد یعنی داده‌ها را از طریق دریافت برنامه به حالت خوانا تبدیل می‌کند
۲. تعریف و پیاده‌سازی قوانین فرآیند ارسال و دریافت پیام مانند فرصت پاسخ، تعداد تلاش‌های ارسال دوباره، جایگزینی، و بسیاری از استثناهای دیگر و قوانین کنترل کننده خطا را شامل می‌شود.
۳. شکل ۲-۳ پیاده‌سازی پروتکل پیام را توسط ماژول پیوند پردازشگر برنامه پرداخت نشان می‌دهد.



شکل ۲-۳: تبادل مابین پروتکل‌ها در پیوند پردازنده را نشان می‌دهد

پروتکل‌های ارسال و دریافت پیام استاندارد در مقابل اختصاصی

پروتکل‌های پیام برای هر تبادل داده‌ای مابین برنامه پرداخت و پردازشگر اخیرا به عنوان قانون توسعه داده شده است. درحالی‌که یک استانداردسازی محکم و مرتبی از شکل کارت پرداخت وجود دارد (ویژگی‌های فیزیکی نوار مغناطیسی و شکل مسیرهای مغناطیسی)، عدم وجود چنین استاندارد مشابهی در ناحیه پروتکل‌های پیام به‌وضوح مشخص است. برخلاف این حقیقت که استاندارد صنعتی رسمی برای پیام‌های تراکنش موجود است (ISO8583)، بیشتر پردازشگرها پروتکل‌های اختصاصی خودشان را ایجاد کرده‌اند، در برخی مواقع فقط نسخه جدیدی از ISO8583 است اما اغلب قوانین و فرمت‌های کاملا متفاوت خاص خود را دارد. چندین نمونه عمومی شده پروتکل‌های پیام (اکثرا منسوخ شده) وجود دارد که بصورت برخط می‌توان آن‌را پیدا کرد.

حقیقت این‌که بسیاری از پردازشگرهای پرداخت هنوز هم مشخصات محرمانه خودشان را حفظ کرده‌اند که یک نمونه بارزی از امنیت در لابه‌لای ابهام است.

بسیاری از پروتکل‌های ارسال و دریافت پیام در دهه ۹۰ میلادی طراحی شده‌اند، و بیشتر آن‌ها بدون هیچ امنیتی ایجاد شده‌اند. تنها نگرانی اعتمادپذیری و مقیاس‌پذیری بود. هیچ عملکرد داخلی برای حفاظت از محرمانگی و درستی (مانند رمزنگاری ، هویت‌سنجی رمزنگاری یا امضای دیجیتال) داده وجود نداشت. وقتی بحث امنیت به‌وجود آمد، فروشندگان پردازشگرها و برنامه پرداخت اکثرا به دو فاکتور زیر اتکا کردند:

۱. امنیت در ابهام که براساس این فرضیه است که اکثر پیوندهای مربوط به پردازشگرها از طریق خطوط اجاره‌ای نگه‌داری می‌شود که تصور می‌شود که خارج از دسترس افراد غیرمجاز باشد.
۲. امنیت ارتباطات که توسط زیرساخت شبکه به‌جای سخت‌افزار یا نرم‌افزار پرداخت ارائه شده است (مانند VPN یا IPSec)

معایب آشکار هر در رهیافت گفته شده این است که داده‌های حساس در یک فایل متن رمز نشده انتقال یافته و براحتی می‌تواند قطع شود.

پروتکل‌های داخلی

همانطور که قبلا در این فصل گفته شد، سه نوع اصلی رابط پرداخت API پایانه فروش، پیوند پردازشگر پرداخت، و رابط دستگاه POI وجود دارد. هرسه نوع رابط‌های ارتباطی خارجی هستند. هرچندکه، برنامه پرداخت ممکن است معماری توزیع یافته‌ای در ماژول‌های جداگانه داشته باشند که در موقعیت‌های فیزیکی جداگانه مستقر شده است. در این موقع، یک رابط ارتباطی داخلی وجود خواهد داشت که داده‌ها را (شامل اطلاعات حساس صاحب کارت) مابین ماژول‌های مختلف کل برنامه پرداخت انتقال می‌دهد. چنین کانال داخلی معمولا پروتکل‌های ارتباط و پیام اختصاصی و اغلب ثبت نشده را بکار می‌گیرند که با پروتکل‌های خارجی متفاوت است.

ارتباط داخلی نسبت به ارتباط خارجی خود می‌تواند آسیب‌پذیرتر باشد چون این ارتباط داخلی در شبکه محلی محافظت شده قرار دارد، از آنجایی‌که PCI DSS و PA-DSS مستلزم رمزگذاری ترافیک LAN نیست.

گسترش برنامه‌های پرداخت

ماژول‌های برنامه پرداخت می‌تواند مابین موقعیت‌های فیزیکی مختلف برای مثال، برروی دستگاه پایانه فروش و BO در فروشگاه و حتی در سرورهای از راه دور اجرا شده مرکز داده گسترش یابد. مدل گسترش، میزان آسیب‌پذیری سیستم پرداخت را معین می‌کند.

مفهوم EPS

در این بخش درباره سیستم پرداخت الکترونیکی یا Electronic payment Service بحث می‌کنیم. برخی اوقات، به سومین کلمه این اختصار کلماتی از قبیل سرور یا سرویس اشاره می‌شود که تغییر چشم‌گیری در معنی اصطلاح نمی‌کند. هدف اصلی سیستم پرداخت الکترونیکی این است که برنامه پردازش پرداخت الکترونیکی را از مابقی عملکردهای پایانه فروش ایزوله نگه دارد. بسیاری از افراد حرفه‌ای عقیده دارند که

فرآیند پرداخت یک دامنه جداگانه و تجارت خود بسنده است. بدون شک پرداخت بخش کامل پایانه فروش چرخه تراکنش است.

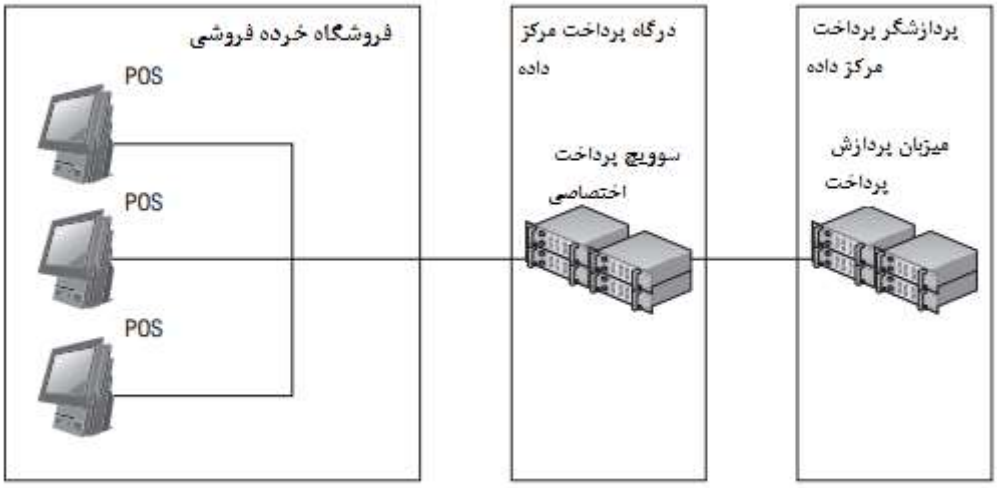
برخلاف این حقیقت که برای معنی آکادمیکی سیستم پرداخت الکترونیکی معنی واضحی وجود ندارد، و مرز مابین عملکرد بالقوه پایانه فروش و فرآیند پرداخت مبهم است، از نظر امنیتی مزایای آشکاری برای چنین جداسازی وجود دارد. فروشندگان کسانی هستند که نرم‌افزار سیستم پرداخت الکترونیکی را ایجاد کرده‌اند و در تکنولوژی‌های پرداخت حرفه‌ای محسوب می‌شوند (شامل امنیت)، درحالی‌که توسعه دهندگان پایانه فروش باید در زمینه های زیادی حرفه‌ای باشند، در نتیجه آنها نمی‌توانند بطور کامل بر روی امنیت تمرکز کنند. یک جداسازی منطقی (و اغلب فیزیکی) از پایانه فروش و سیستم پرداخت، اجازه حذف پایانه فروش را از مقیاس مورد نظر را می‌دهد (نویسندگان واژگان امنیتی بدین طریق معنی می‌کنند که الزامات استاندارد امنیتی از قبیل PCI با یک برنامه یا دستگاه خاصی متناسب نیستند).

قرار دادن برنامه یا دستگاه پایانه فروش در خارج از محدوده، چندین توسعه و کار پیاده سازی را برای تولیدکنندگان نرم‌افزار و مصرف‌کنندگان را صرفه جویی می‌کند. هر چند که، سیستم پرداخت الکترونیکی یک راه‌حل تمام عیار نیست و فاکتورهایی دیگری نیز موجود است مانند، گسترش مدل و طرح رمزنگاری که امنیت کلی سیستم را تحت تاثیر قرار می‌دهد.

سووئیچ پرداخت

حداقل دو نوع عملکرد رایانه پایانه فروش و BO در فروشگاه خرده فروشی وجود دارد. ماژول‌های برنامه پرداخت می‌توانند مابین این دونوع توزیع شوند که وابسته به طراحی خاص فروشنده نرم‌افزار است یا گاهی اوقات وابسته به الزامات پردازشگر پرداخت تاجرها است. در حقیقت، یک موقعیت محتمل دیگری در بیرون از فروشگاه (سوم) وجود دارد.

بسیاری از فروشندگان نرم‌افزار پرداخت یا تاجرها میزبان پردازش‌کننده خود را دارند که پیام‌های تراکنش حاصله از چندین فروشگاه را جمع‌آوری و آن‌ها را به پردازشگر متناسب پرداخت ارسال می‌کند. این میزبان معمولا با نام سووئیچ پرداخت شناخته می‌شود. عملکرد اصلی آن جدا کردن برنامه پرداخت از قسمت‌هایی است که در مرکز بهتر می‌توانستند انجام شوند، برای مثال، هدایت تراکنش به یک پردازشگر متناسب مبتنی بر BIN یا نوع کارت (شکل ۲-۴).



شکل ۲-۴: سوییچ پرداخت

مقایسه مدل‌های گسترش

محاسبه آسیب‌پذیری هر یک از مولفه برنامه پرداخت بر اساس فاکتورهای بدست آمده از نواحی حساس آسیب‌پذیر تعریف شده در فصل اول بدست آمده است: (جدول ۲-۳)

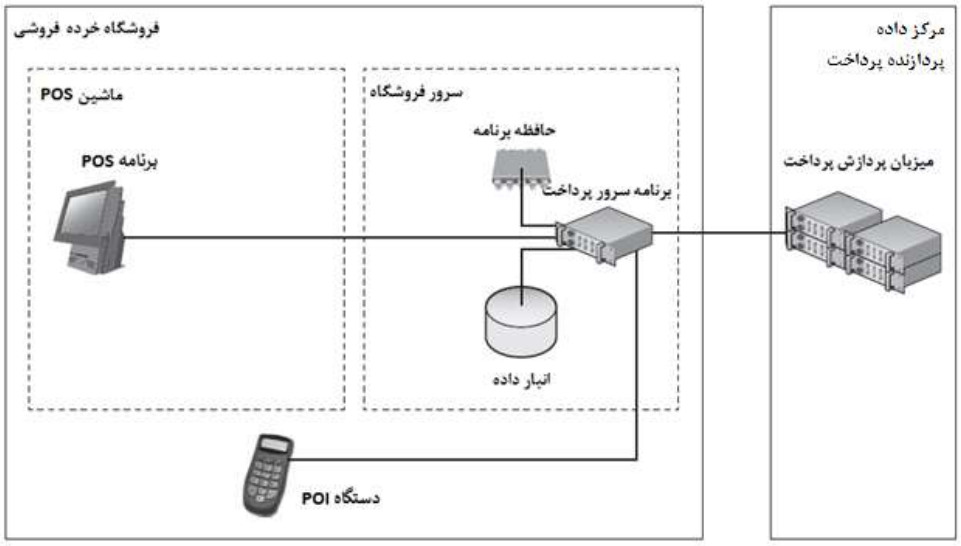
- داده در حافظه
- داده ذخیره شده
- داده در حال انتقال
- تنظیمات و کد برنامه

جدول ۲-۳: فاکتورهای محاسبه آسیب‌پذیری

نواحی در معرض	نواحی حساس آسیب‌پذیر
حافظه	داده‌های حافظه
ذخیره موقت داده	داده غیر فعال
سوابق توافق	داده غیر فعال
کد و تنظیمات برنامه	کد و تنظیمات برنامه
اتصال به پایانه فروش	داده در حال انتقال
ارتباطات داخلی	داده در حال انتقال
پیوندهای میزبان	داده در حال انتقال

مدل استقرار سیستم پرداخت الکترونیکی فروشگاه

همانطور که در شکل ۲-۵ نشان داده شده است در مدل استقرار سیستم پرداخت الکترونیکی فروشگاه، پردازش پرداخت توسط سیستم پرداخت الکترونیکی در مرکز و سرور فروشگاه انجام می‌شود. سیستم پرداخت الکترونیکی ارتباط مستقیم با همه دستگاه‌های پایانه فروش و POI را نگه می‌دارد. سیستم پرداخت الکترونیکی همه چیز را از جریان ورود کارت تا تراکنش‌های پرداخت پردازش را مدیریت می‌کند. بنابراین، هیچ داده حساسی بین پایانه فروش و سیستم پرداخت الکترونیکی انتقال نمی یابد.



شکل ۲-۵: مدل استقرار سیستم پرداخت الکترونیکی فروشگاه

مزایا و معایب امنیتی این مدل عبارت است از:

- مزایا—ماشین POI در برابر داده‌های حساس قرار ندارد. زیرا ارتباطی با دستگاه‌های POI ندارد.
- مزایا—ارتباط بین ماشین‌های سرور فروشگاه و پایانه فروش شامل داده‌های حساس نیست. بنابراین این ترافیک نیازی به رمزنگاری ندارد.
- معایب—ارتباط بین دستگاه‌های POI و سرور فروشگاه از طریق شبکه محلی (معمولاً بسته‌های TCP/IP) فروشگاه پیاده‌سازی می‌شود، اطلاعات حساس صاحبان کارت در معرض شبکه قرار می‌گیرند.

داده ورودی و نتایج محاسبه امتیاز آسیب‌پذیری برای مدل استقرار سیستم پرداخت الکترونیکی فروشگاه در جدول ۲-۴ نشان داده می‌شود.

جدول ۲-۴: امتیاز آسیب‌پذیری مدل استقرار سیستم پرداخت الکترونیکی فروشگاه

ناحیه در معرض قرار گرفته	عدم الزام در حفاظت از برنامه (X2)	آسیب‌پذیری در ماشین POS (+1)	آسیب‌پذیری در سرور فروشگاه (+1)	امتیاز
حافظه	•	–	•	۲
ذخیره‌سازی موقت داده	–	–	•	۱
ثبت واریزها	–	–	•	۱
پیکربندی و کد برنامه	•	–	•	۲
ارتباط با POI	•	–	•	۲
ارتباط‌های داخلی	•	–	–	۰
پیوندهای میزبان	•	–	•	۲
امتیاز آسیب‌پذیری				۱۰

مدل استقرار سیستم پرداخت الکترونیکی پایانه فروش

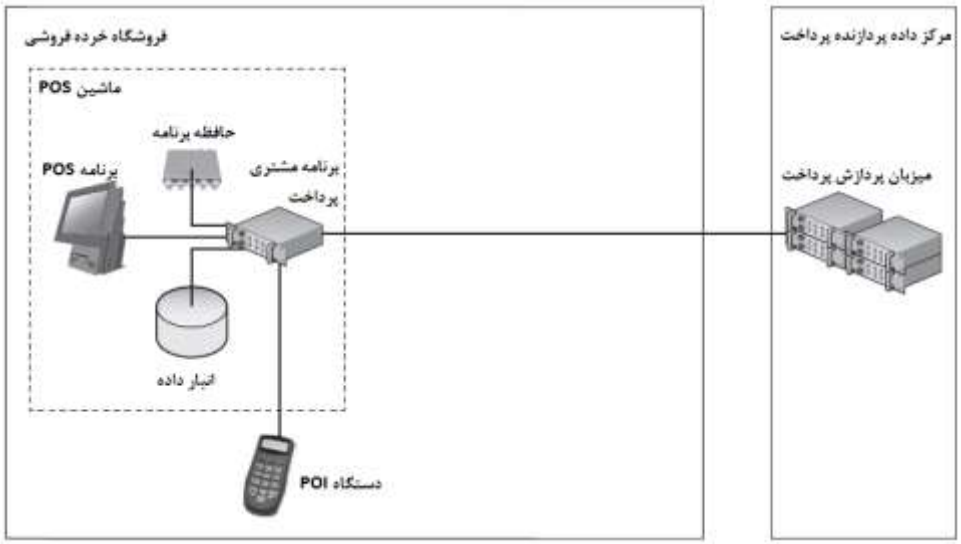
همان‌طور که در شکل ۲-۶ نشان داده شده است در مدل استقرار سیستم پرداخت الکترونیکی پایانه فروش، پردازش پرداخت توسط سیستم پرداخت الکترونیکی که در هر ماشین پایانه فروش قرار گرفته، انجام می‌شود. سیستم پرداخت الکترونیکی ارتباط مستقیم با پایانه فروش و دستگاه‌های POI را نگه می‌دارد و همه چیز از جریان ورود کارت تا تراکنش پرداخت پردازش را مدیریت می‌کند. بنابراین، هیچ داده حساسی بین پایانه فروش و سیستم پرداخت الکترونیکی انتقال نمی‌یابد. سیستم پرداخت الکترونیکی ارتباط مستقیم با سویچ پردازنده پرداخت که خارج فروشگاه قرار دارد، را حفظ می‌کند.

مزایا و معایب این مدل عبارت است از:

- مزایا—هیچ مکان مرکزی در فروشگاه وجود ندارد که همه داده‌های حساس در حافظه، دیسک دخیره‌سازی یا ترافیک شبکه را جمع آوری کند. حفاظت از یک ماشین منفرد و نمونه برنامه آسان‌تر (و کم هزینه) است. هرچند اگر خراب شود، تمام داده‌های قروشگاه از بین می‌رود.
- مزایا— (کد) برنامه پایانه فروش داده‌های حساس را مدیریت نمی‌کند زیرا تمام عملکرد پرداخت به یک برنامه جداگانه سیستم پرداخت الکترونیکی واگذار می‌شود.

- تمام ماشین‌های پایانه فروش (حافظه، ذخیره‌سازی داده‌ها) در فروشگاه در معرض داده‌های حساس و همچنین ارتباط بین ماشین پایانه فروش و میزبان پرداخت قرار می‌گیرد.

داده‌های ورودی و نتایج امتیاز آسیب‌پذیری برای مدل استقرار سیستم پرداخت الکترونیکی پایانه فروش در جدول ۲-۵ ذکر شده است.



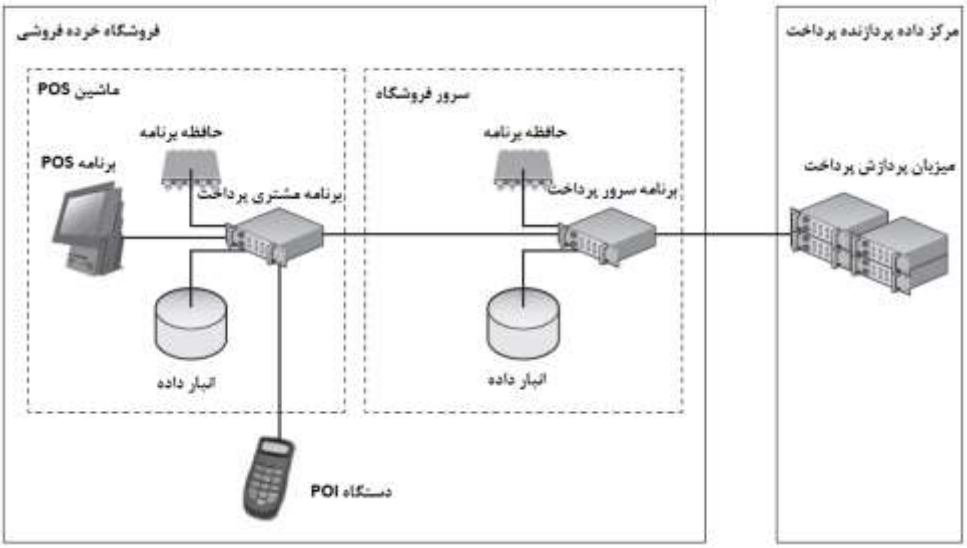
شکل ۲-۶: مدل استقرار POS EPS

جدول ۲-۵: امتیاز آسیب‌پذیری مدل استقرار EPS POS

ناحیه در معرض قرار گرفته	عدم الزام در حفاظت از برنامه (X2)	آسیب‌پذیری در ماشین POS (+1)	آسیب‌پذیری در سرور فروشگاه (+1)	امتیاز
حافظه	•	•	-	۲
ذخیره‌سازی موقت داده	-	•	-	۱
ثبت واریزها	-	•	-	۱
پیکربندی و کد برنامه	•	•	-	۲
ارتباط با POI	•	•	•	۲
ارتباط‌های داخلی	•	-	-	۰
پیوندهای میزبان	•	•	•	۲
امتیاز آسیب‌پذیری				۱۰

مدل استقرار پایانه فروش/فروشگاه ترکیبی

مدل استقرار پایانه فروش/فروشگاه ترکیبی از آسیب‌پذیرترین روش‌ها محسوب می‌شود، زیرا ماژول‌های برنامه پرداخت در سراسر ماشین‌های فیزیکی مختلف گسترش می‌یابند. همان‌طور که در شکل ۲-۷ نشان داده شده است، پردازش اولیه پرداخت (مانند تعامل با POI ومدیریت جریان تراکنش پرداخت) در ماشین پایانه فروش انجام می شود و همچنین با ماژول سرور در سطح فروشگاه ارتباط برقرار می‌کند. پیوندها در سویچ پرداخت یا پردازنده‌های سرور فروشگاه اجرا می‌شود.



شکل ۲-۷: مدل استقرار POS/Store ترکیبی

هیچ فواید امنیتی در این مدل وجود ندارد. ایراد امنیتی این است که هر دو ماشین پایانه فروش و ماشین سرور فروشگاه و تقریبا تمام اجزای آن (حافظه، ذخیره‌سازی داده‌ها، کد برنامه، و خطوط ارتباطی) کاملا آسیب‌پذیر هستند.

داده‌های ورودی و نتایج محاسبه امتیاز آسیب‌پذیری برای مدل استقرار سیستم پرداخت الکترونیکی پایانه فروش/فروشگاه ترکیبی در جدول ۲-۶ آورده شده است.

جدول ۲-۶: امتیاز آسیب‌پذیری مدل استقرار POS/Store ترکیبی

ناحیه در معرض قرار گرفته	عدم الزام در حفاظت از برنامه (X2)	آسیب‌پذیری در ماشین POS (+1)	آسیب‌پذیری در سرور فروشگاه (+1)	امتیاز
حافظه	•	•	•	۴
ذخیره‌سازی موقت داده	-	•	•	۲

ثبت واریزها	–	–	•	۱
پیکربندی و کد برنامه	•	•	•	۴
ارتباط با POI	•	•	-	۲
ارتباط‌های داخلی	•	•	•	۴
پیوندهای میزبان	•	–	•	۲
امتیاز آسیب‌پذیری				۱۹

سیستم‌های پرداخت جایگاه بنزین

استقرار یک سیستم پرداخت در یک جایگاه بنزین متفاوت از آنچه که در یک فروشگاه معمولی وجود دارد، می‌باشد. زیرا شامل قسمت‌های سخت‌افزاری و نرم‌افزاری اضافی مانند جایگاه سوخت‌گیری و کنترل‌گرهای پیشرفته جایگاه سوخت است. همانند سخت‌افزار پایانه فروش و POI ، بسیاری از مدل‌های پمپ وجود دارد که توسط چندین فروشنده مختلف تولید می‌شوند. از دیدگاه امنیت پرداخت، می‌توان آنها را حداقل به دو گروه پایانه‌های پرداخت ناظر[۲۹] و پایانه‌های پرداخت منطقه‌ای[۳۰] تقسیم کرد.

تفاوت این دو ساده است—اولی دارای دستگاه‌های صفحه رمز و نوار مغناطیس‌خوان که داخل جایگاه قرار دارد، می‌باشد در حالی که دومی دارای دستگاه‌های POI مستقلی است که به‌طور منطقی از جایگاه و کنترل‌گر جداست (هرچند مدل‌های پایانه‌های پرداخت ناظر وجود دارند که به‌طور منطقی از جایگاه جدا می‌شوند). بدیهی است نوع دوم امن‌تر است. زیرا اجازه می‌دهد مفهوم سیستم پرداخت الکترونیکی با تفکیک کامل بین پایانه فروش و جریان‌های پرداخت پیاده‌سازی شود.

اولین گروه – پایانه پرداخت یکپارچه به دو دلیل آسیب‌پذیر است:

۱. داده‌های حساس صاحبان کارت در معرض قسمت‌های سخت‌افزار و نرم‌افزار مانند کنترل‌گر جایگاه سوخت و خطوط ارتباطی اضافی قرار می‌گیرند.
۲. در حال حاضر، اکثر کارت‌خوان‌های ساخته شده و کنترل‌گرهای جایگاه سوخت، سخت‌افزار P2PE را پشتیبانی نمی‌کنند.

علاوه بر این، جایگاه‌های سوخت‌گیری اغلب بدون مراقب هستند که به وضوح به امنیت صفحه‌های رمزشان کمک نمی‌کند.

---

[۲۹] *Dispenser Payment Terminals* (DPT)
[۳۰] *Island Payment Terminals* (IPT)

این موضوع نشان می‌دهد که برنامه‌های پرداخت جایگاه بنزین حتی می‌تواند خطرناک‌تر از همتایان خود در فروشگاه‌های خرده‌فروشی معمولی باشد.

پرداخت‌های سیار

امروزه، هنگام صحبت در مورد پرداخت‌های الکترونیکی، اشاره نکردن به امیدوارکننده‌ترین گرایش صنعت یعنی پرداخت‌های سیار غیر ممکن است. با وجود اینکه پرداخت با تلفن همراه در فروشگاه‌های هنور یک روند رایج در ایالات متحده نیست، و ممکن است قبل از این که یک روند رایج شود، فقط یک موضوع در سال‌های اخیر باشد. بسیاری از پردازنده‌های پرداخت، فروشندگان نرم‌افزار پرداخت، ارائه دهندگان خدمات مخابراتی تلفن همراه، بانک‌ها، برندهای پرداخت، شرکت‌های پرداخت و حتی موتورهای جستجوی اینترنتی برای اولین بودن در پذیرش پرداخت‌های سیار و تبدیل شدن به استاندارد در این بخش پیش‌رو تلاش می‌کنند. همیشه فناوری جدید، چالش‌های امنیتی جدیدی رو به همراه دارد.

درحال حاضر، دو نوع اصلی از فناوری پرداخت سیار وجود دارد:

۱. مبتنی بر ارتباط میدان نزدیک[۳۱]
۲. هر چیز دیگر

هر دو رویکرد مشکلات مشابهی با ویژگی‌های امنیت برنامه وب و امنیت سیار دارند. هرچند، برنامه‌های پرداخت مبتنی بر ارتباط میدان نزدیک آسیب‌پذیری‌های بیشتری در ارتباط با نحوه ارتباط یک دستگاه سیار با پایانه فروش تاجر دارند. معماری و مدل استقرار روش‌های پرداخت سیار معمولی برای فروشگاه‌های خرده‌فروشی را در ادامه مرور می‌کنیم.

فناوری‌های پرداخت سیار مبتنی بر ارتباط میدان نزدیک

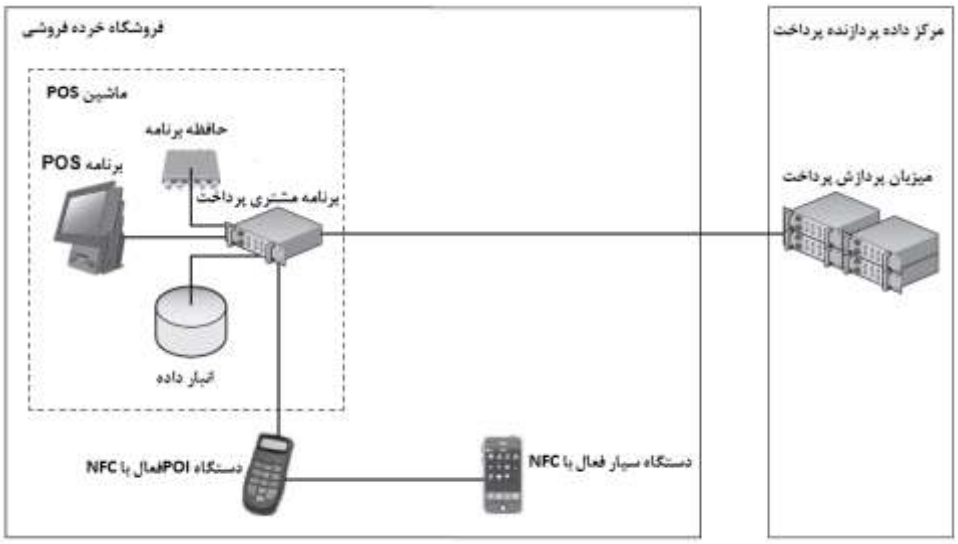
ویژگی معمول این گروه از روش‌های پرداخت سیار که از فناوری ارتباط میدان نزدیک بهره می‌برند این است که اجازه می‌دهد تبادل اطلاعات بین دستگاه‌های الکترونیکی در محدوده بسیار نزدیک با استفاده از ارتباطات رادیوئی بسامد بالا(۱۳,۵۶ مگاهرتز) انجام پذیرد. ارتباط میدان نزدیک که اساساً مجموعه‌ای از چندین پروتکل و استاندارد است؛ توسط چندین برنامه مختلف از جمله کارت‌های محدوده، برچسب‌های ارتباط میدان نزدیک و اخیراً پرداخت‌های سیار استفاده می‌شوند. روش‌های پرداخت سیار مانند کیف پول گوگل وجود دارند که از ویژگی‌های منحصربفرد ارتباط میدان نزدیک (جدول ۲-۷) مانند حداکثر محدوه کوتاه (کمتر از چند اینچ) و زمان راه‌اندازی اتصال بسیار سریع(<0.1s) بهره می‌برند.

جدول ۲-۷: مقایسه NFC با دیگر فناوریهای ارتباطات بدون تماس

[۳۱] Near Field Communication (NFC)

فناوری	محدوده معمول	زمان راه‌اندازی اتصال	خارج از محدوده رمزگذاری
NFC	1 ~ اینچ	کمتر از 0.1 ثانیه	خیر
Bluetooth	۳۰ فوت	کمتر از ۶ ثانیه	بله
Wi-Fi	۱۲۰ فوت	چندین ثانیه	بله

در چنین روش‌هایی، یک تلفن سیار مجهز به یک دستگاه NFC، یک کارت پرداخت بدون تماس را شبیه-سازی می‌کند که می‌تواند توسط یک POI فعال شده با NFC خوانده شود (شکل ۲-۸ را ببینید).



شکل ۲-۸: معماری و استقرار یک روش پرداخت مبتنی بر NFC

داده‌های تأیید هویت کارت پرداخت (محتوای نوارهای مغناطیسی کارت‌های پلاستیکی) می‌توانند یا برروی دستگاه سیار دخیره شوند یا از ابر دانلود گردند. در هر صورت، داده‌های حساس باید از دستگاه سیار به دستگاه POI به ترتیب منتقل شوند تا تراکنش پرداخت آغاز شود که زمینه را برای حملات بالقوه فراهم می‌کند و نگرانی‌های امنیتی را افزایش می‌دهد. این نگرانی‌ها در فصل آخر بحث می‌شوند.

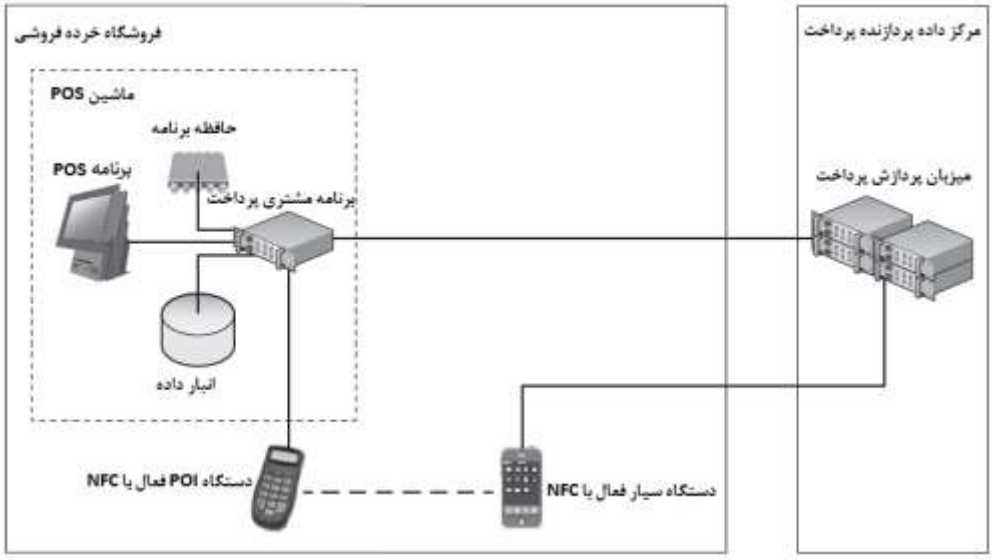
روش‌های پرداخت سیار غیرNFC

بسیاری از روش‌های پرداخت سیار جایگزین وجود دارند که مبتنی برNFC نیستند. چنین برنامه‌هایی دارای مزایای قوی نسبت به محصولات مبتنی بر NFC هستند. زیرا آنها به تجهیزات NFC نیازی ندارند. که این خود یک مزیت حساب می‌شود چون:

- بسیاری از تلفن‌های سیار مشهور مانند Apple iPhone به فرستنده NFC مجهز نیستند.
- تنها تعداد محدودی از تاجران دستگاه‌های POI را به NFCخوان‌ها توسعه داده‌اند.

یک نمونه از برنامه‌های پرداخت سیار "غیر-NFC" برنامه سیار Starbuckها است که شماره کارت را به خوبی یک بارکد بر روی صفحه گوشی نمایش می‌دهد. برای شروع پرداخت، شماره کارت بصورت دستی وارد می‌شود یا بارکد توسط صندوق‌دار اسکن می‌شود و پرداخت از این نقطه همانند یک پرداخت کارت معمول پردازش می‌شود.

اگر یک برنامه پرداخت سیار با توجه به امنیت طراحی شده باشد، به جای نگهداری شماره کارت در دستگاه و نمایش آن بر روی صفحه، بطور تصادفی یک بار رمزی را که برای ایجاد ارتباط بین دستگاه سیار مشتری و POS تاجر نمایش داده می‌شود (و داخل بارکد رمزنگاری می‌شود) تولید می‌کند (شکل ۲-۹). داده‌های احراز هویت شده واقعی (مانند PAN و تاریخ انقضا) در ابر بین مراکز داده روش سیار و مراکز داده پردازنده پرداخت (یا حتی داخل همان مرکز داده، اگر پردازنده پرداخت، ارائه دهنده روش پرداخت سیار باشد) منتقل می‌شود.



شکل ۲-۹: معماری و استقرار امنیتی پرداخت‌های سیار غیر NFC

در یک روش پرداخت سیار "غیر NFC" با طراحی خوب، هرگز داده‌های حساس احراز هویت شده را با دستگاه سیار و برنامه پرداخت POS ارتباط نمی‌دهد، که یک مزیت امنیتی بزرگ در مقایسه با مدل NFC بدون تماس است که در آن انتقال داده‌های واقعی کارت از دستگاه سیار به POS نیاز است.

خلاصه فصل

برنامه‌های پرداخت معمول شامل چندین واسطهای کاربری و ماژول‌های پردازشی است:

- رابطها
- دستگاه POI
- API پایانه فروش

- پیوند پردازنده پرداخت
- ماژول‌های پردازشی
  - مسیریاب
  - ذخیر و ادامه
  - TOR
  - مجموعه دستور

رابط‌های کاربری با استفاده از انواع مختلف فناوری‌های اتصالی، ارتباطی و پروتکل‌های پیغام با دنیای بیرون ارتباط دارند. ماژول‌های پردازشی جریان کار را اجرا می‌کنند و داده‌های حساس صاحبان کارت را برروی دیسک‌های سخت ذخیره می‌کنند. هیچ فناوری امنیتی (رمزنگاری) تعریف شده برای پروتکل‌های رابط کاربری (داده در موقع ارسال) و یا ماژول‌های پردازشی (داده ذخیره شده) برنامه‌های پرداخت وجود ندارد.

سه مدل اصلی استقرار برنامه پرداخت وجود دارد:

۱. سیستم پرداخت الکترونیکی فروشگاه
۲. سیستم پرداخت الکترونیکی پایانه فروش
۳. Store/POS ترکیبی

هر مدل مزایا و معایب امنیتی خود را دارد. هرچند، مدل Store/POS ترکیبی آسیب‌پذیرترین مدل است.

روش‌های پرداخت سیار به دو گروه اصلی تقسیم می‌شوند:

۱. مبتنی بر NFC
۲. غیر NFC

برنامه‌های مبتنی بر NFC بطور بالقوه آسیب‌پذیرتر هستند، زیرا داده‌های حساس صاحبان کارت را ذخیره می‌کنند و یا انتقال می‌دهند.

فصل سوم:

PCI

استانداردها یک پدیده جالب خصوصاً در زمینه فناوری اطلاعات هستند. از یک سو آنها تشریفات اداری ایجاد می‌کنند، خلاقیت را از بین می‌برند، و بسیاری ازافراد بااستعداد را می‌ترسانند. از سوی دیگر، استانداردها منابع را ذخیره می‌کنند، قابلیت اطمینان فراهم می‌کنند و اجازه می‌دهند افراد و سازمان‌های مختلف با زبان یکسان با یکدیگر تعامل کنند.

PCI چیست؟

روش اصلی استانداردهای امنیتی PCI، با ایجاد لایه‌های بیشتری از کنترل‌های امنیتی در فن‌آوری‌های موجود، آسیب پذیری سیستم‌های پرداخت الکترونیکی را جبران می‌کند. این امر، کاربران نهایی (پردازنده‌های پرداخت، ارائه‌دهندگان خدمات، فروشندگان سخت‌افزار/نرم‌افزار و در نهایت تاجران) را پاسخگو می‌سازد؛ بنابراین، سیستم‌های پرداخت اغلب ناامن هستند حتی اگر سازگار با PCI باشند.

استانداردهای PCI

استانداردهای PCI جنبه‌های مختلف از چرخه پرداخت الکترونیکی را پوشش می‌دهند. نخستین و معروف‌ترین بخش، استاندارد امنیتی داده PCI (PCI DSS) است که به تاجران (مانند خرده‌فروشان) و ارائه‌دهندگان خدمات (مانند دروازه‌ها و پردازنده‌های پرداخت) می‌گوید چگونه از اطلاعات حساس دارنده‌کارت محافظت کنند. دومین و مهم‌ترین بخش، استاندارد امنیتی داده‌های برنامه پرداخت (PA-DSS) برای فروشندگان نرم‌افزار پرداخت است که می‌گوید چگونه باید محصولات خود را طراحی کنند تا با استانداردهای PCI DSS سازگاز باشد. بخش سوم PTS است که از ابزارهای سخت افزاری نظیر POI و HSMو همچنین ماژول‌های رمزنگاری و نرم‌افزار دائمی برنامه‌نویسی شده در حافظه فقط خواندنی محافظت می‌کند و در نهایت، P2PE که پیچیدگی سه بخش قبلی را دربرگرفته و چالش‌های جدیدی را در قالب نیازهای صحت سنجی نرم افزاری و سخت افزاری بیشتر که قبلا ناشناخته بود؛ اضافه می‌کند. جدول ۳-۱ کاربرد چهار استاندارد PCI را نشان می‌دهد.

جدول ۳-۱: قابلیت‌های اجرائی استانداردهای PCI

استاندارد PCI	اشیا	اهداف	موضوع
PCI DSS	محیط	محیط پردازش تراکنش پرداخت مانند فروشگاه‌های خرده‌فروشی و مراکز داده فرآیند پرداخت	تاجران، ارائه‌دهندگان خدمات پرداخت، بانک‌ها) (پردازنده‌های پایانه‌ها،

PA-DSS	نرم‌افزار	برنامه‌های پرداخت	فروشندگان نرم‌افزار
PTS	سخت‌افزار	SCD (شامل نرم‌افزارهای برنامه‌نویسی شده در حافظه فقط خواندنی): POI، HSM	سازندگان سخت‌افزار
P2PE	محیط، سخت‌افزار، نرم‌افزار	روش‌های P2PE، برنامه‌های P2PE	ارائه‌دهندگان روش P2PE، فروشندگان نرم‌افزار P2PE

از آنجا که همواره سردرگمی و سوالات بسیاری در مورد قابلیت اجرائی استانداردهای PCI خاص وجود دارد، (جدول ۳-۲) در این موضوع از زوایای مختلف در نظر می گیریم.

جدول ۳-۲: تطابق با استانداردهای PCI

اگر شما	شما باید منطبق باشید با ....
توسعه‌دهنده و فروشنده برنامه‌های پرداخت	PA-DSS
خریدار و استفاده‌کننده از برنامه پرداخت	PCI DSS
فروشنده مجدد و راه‌انداز برنامه‌های پرداخت	-
توسعه‌دهنده برنامه پرداخت برای مشتری منفرد	-
توسعه‌دهنده و استفاده‌کننده از برنامه‌های پرداخت	PCI DSS
ارائه‌دهنده برنامه پرداخت به عنوان خدمت	PA-DSS and/or PCI DSS
پردازش‌کننده تراکنش‌های پرداخت	PCI DSS
سازنده دستگاه POI	PTS
سازنده دستگاه HSM	PTS
توسعه‌دهنده و فروشنده برنامه‌های پرداخت P2PE	P2PE
توسعه‌دهنده و ارائه‌کننده روش P2PE به عنوان خدمت	P2PE
هک‌کننده برنامه‌های پرداخت	استانداردهای اخلاقی مربوط به خود

PA-DSS در مقابل PCI DSS

تفاوت واقعی بین PCI DSS و PA-DSS اشیا، اهداف و موضوع نیست، بلکه مسئولیت‌پذیری است: چه کسی فقدان امنیت کارت اعتباری را جبران می‌کند و سرعت حمله را می‌کاهد؟ چنین مسئولیتی بین تاجران و فروشندگان نرم‌افزار بدون هیچ نسبت معقولی، با بار اصلی بر روی تاجران، به اشتراک گذاشته می‌شود. جدول ۳-۳ خطسیر تعرض سطح بالا و مسئولیت‌پذیری برای کاستن آنها، را بررسی می‌کند.

جدول ۳-۳: مسؤلیت اجرای کنترل‌های امنیتی بر اساس PCI

ناحیه‌های آسیب‌پذیری	واقع در	مسؤل جبران
حافظه برنامه پرداخت	PCI DSS	تاجر
ارتباط در کل شبکه محلی	PCI DSS	تاجر
ارتباط در کل شبکه عمومی	PCI DSS ، PA-DSS	تاجر-فروشنده برنامه پرداخت
ارتباط بین PA و POI	PCI DSS	تاجر
پیوند با پردازنده‌پرداخت (اگر از راه شبکه عمومی نیست)	PCI DSS	تاجر
کد برنامه	PCI DSS، PA-DSS	تاجر
تنظیمات برنامه	PCI DSS	تاجر
دستگاه POI (فیزیکی)	PCI DSS	تاجر
مدیریت کلید رمزنگاری	PCI DSS، PA-DSS	تاجر
انبار داده‌های حساس	PCI DSS، PA-DSS	تاجر، فروشنده برنامه پرداخت
مسؤلیت کلی برای نقض داده‌های کارت	PCI DSS	تاجر

همانطور که در جدول ۳-۳ مشاهده می‌کنید، حتی حفاظت از ناحیه‌هایی مانند حافظه یا کد برنامه، که با نرم‌افزار مرتبط است، تحت مسئولیت کاربر (تاجر) قرار می‌گیرد. PA-DSS به توسعه‌دهندگان نرم‌افزار نیازی ندارد که اطلاعات حساس در حافظه را رمزنگاری کند یا کدکامپایل شده خود را مبهم سازد. تاجران باید

برنامه های پرداخت را از طریق اجرای انواع کارهای مختلف مانند دیوارآتش و مانیتورینگ یکپارچگی فایل محافظت کنند.

PA-DSS

استاندارد امنیتی داده‌های برنامه پرداخت (PA-DSS) به عنوان یک برنامه صحت‌سنجی برای فروشندگان نرم‌افزار پایانه فروش ایجاد شده بود. کسانی که محصولات خود را برای چندین مشتری به فروش می-رسانند؛ این اطمینان را برای خریداران فراهم می‌کنند که خرید جدید آنها قوانین PCI DSS را نقض نمی‌کند. PA-DSS تنها یکی از استانداردهای خانواده PCI است که به طور مستقیم از امنیت برنامه پرداخت محافظت می‌کند.

فرآیند صحت‌سنجی

به طور خلاصه، فرآیند صحت‌سنجی شامل عبور از روش‌های تست (۱۶۸ در PA-DSS) است. به محض این که فروشنده برنامه پرداخت ارزیابی امنیتی مناسب[۳۲] (QSA) را از فهرست منتشر شده در وب سایت PCI SSC 2 انتخاب می‌کند و قراردادها را امضا می‌کند (و البته هرینه پرداخت می‌کند)؛ ارزیابی شروع می‌شود.

روند صحت‌سنجی خیلی پیچیده نیست و معمولا شامل مراحل زیر است:

۱. جلسه اولیه (می‌تواند در محل و یا با تماس تلفنی باشد.) توسعه‌دهندگان برنامه پرداخت و-PA QSA دامنه پروژه، محدودیت زمان، و طراحی کلی برنامه را بررسی می‌کنند، و در مورد نیازهای آزمایشگاه و اسناد و مدارک بحث می‌کنند.
۲. ارزیابی در محل که خود شامل دو مرحله است:

a. تست برنامه واقعی در آزمایشگاه فروشنده – مامور رسیدگی، PA-QSA، که توسط توسعه‌دهندگان آموزش داده شده است؛ تست نفوذ را با اجرای همه نوع از تراکنش‌های ممکن و کارت ها انجام می‌شود تا اطمینان حاصل شود که هیچ داده‌های حساسی در دیسک سخت نوشته نمی شود. ممکن است برخی از شبکه ها را شنود کنند تا اطمینان حاصل شود که هیچ داده حساس در متن رمزنشده ارسال نمی‌شود. هیچ ابزار یا تکنیک خاصی مورد نیاز نیست و هیچکس سعی در شکستن سیستم ندارد. زیرا QSA یک تست نفوذ حرفه‌ای نیست. در اغلب موارد، اگر برنامه داده‌های حساس را، در متن رمزنشده ذخیره نکند (یا وانمود کند که آن را ذخیره نمی‌کند)، آن آزمون را می‌گذرد.
b. مصاحبه با پرسنل- یک تشریفات است. ماموران رسیدگی باید چند سوال بپرسند و اسامی توسعه‌دهندگان و تیم QA را برای گزارش ثبت کنند.

[۳۲] Qualified Security Assessor

۳. بررسی فرآیندهای پیاده‌سازی، اسناد و مدارک و شواهد-این مرحله بیشتر تشریفات اداری است. دردناک‌ترین قسمت (مخصوصا برای اولین بار)، راهنمای پیاده سازی PCI ایجاد می‌کند که سندی است که هر فروشنده برنامه پرداخت باید به مشتریان (کاربران نرم افزار پرداخت) ارائه دهد. محتوای این سند عمدتا سلب مسئولیت فنی است که به تاجران یادآوری می‌کند که پذیرش کارت اعتباری یک تجارت خطرناک است و مسئولیت تاجر است که معایب طراحی امنیتی پردازش کارت پرداخت را رفع کند.

۴. ایجاد گزارش صحت‌سنجی- این سند توسط PA-QSA تهیه شده و برای پذیرش آن به شورای PCIارسال می‌شود. این مرحله ممکن است در مواقعی که شورا بیانیه‌ای را در گزارش رد کند و آن را به QSA بازگرداند، زمان‌بر باشد. سپس QSA همه قسمت‌های رد شده را به صورت امن‌تر بازنویسی می‌کند و سند را مجدداً ارائه می‌دهد. هنگامی که ROV توسط شورا پذیرفته شده (تایید شده)، فاکتور به فروشنده ارائه می‌شود و برنامه تایید شده، ذکر می‌شود (تا زمانی که صورتحساب پرداخت می‌شود).

لیست برنامه‌های پرداخت تایید شده

شورای PCI لیست برنامه‌های پرداخت تایید شده را در وب سایت خود نگه می‌دارد. قبل از خرید نرم افزار پرداخت جدید و یا نصب یک نسخه جدید، تاجر باید وب سایت را بررسی کند تا اطمینان حاصل شود که محصول تایید شده و واجد شرایط برای استقرار جدید است. صحت‌سنجی مجدد برای هر نسخه جدید برنامه پرداخت یا حداقل بصورت سالانه نیاز می‌شود.

نیازمندی‌های سیزده‌گانه و امنیت پایانه فروش

جدول ۳-۴ بررسی می‌کند که چگونه ۱۳ نیازمندی PA-DSS به توسعه‌دهندگان کمک می‌کند تا نواحی آسیب‌پذیری اصلی برنامه‌های پرداخت را محافظت کنند.

جدول ۳-۴: نیازمندی‌های PA-DSS و نواحی آسیب‌پذیری کلیدی برنامه پرداخت

	نیازمندی PA-DSS	داده حافظه	در داده در انتقال	موقع	داده در محل	تنظیمات و کد AP
۱	عدم نگه‌داری نوار مغناطیسی کامل، کد یا مقدار تأییده کارت (CAV2، CID، CVC2، CVV2) یا داده‌های بلوکی PIN	–	–		○	–
۲	محافظت از داده‌های ذخیره شده دارنده‌کارت	–	–		•	–

۳	فراهم کردن ویژگی‌های احراز هویت امن	–	–	–	–
۴	ثبت فعالیت برنامه پرداخت	–	–	–	–
۵	توسعه برنامه‌های پرداخت امن	–	–	–	○
۶	محافظت از انتقال‌های بی‌سیم	–	○	–	–
۷	تست برنامه‌های پرداخت برای نشان دادن آسیب‌پذیری	–	–	–	○
۸	تسهیل کردن پیاده‌سازی شبکه امن	–	–	–	–
۹	داده‌های دارنده‌کارت نباید بر روی سرور متصل به اینترنت ذخیره شود	–	–	–	–
۱۰	تسهیل کردن دستیابی راه دور امن برای برنامه پرداخت	–	–	–	–
۱۱	رمزنگاری ترافیک حساس در کل شبکه‌های عمومی	–	○	–	–
۱۲	رمزنگاری تمام دسترسی‌های مدیریتی غیر کنسولی	–	–	–	–
۱۳	حفظ اسناد آموزشی و برنامه‌های آموزشی برای مشتری‌ها، فروشندگان و افراد مرتبط با آنها	–	–	–	–

توضیح اینکه:

•: اگرپیاده‌سازی به درستی باشد یک محافظت کامل و کافی ارائه می‌دهد.

○: حفاظت محدود شده و انتخابی را ارائه می‌دهد

– : هیچ‌گونه حفاظت مستقیمی را ارائه نمی‌دهد.

همانطور که در جدول ۳-۴ مشاهده می کنید، این ۱۳ مورد، احتمالا از داده‌های دارنده‌کارت در سرقت از یک یا چند مورد حفاظت نمی‌کند و تنها پوشش کاملی که PA-DSS فراهم می‌کند برای داده در محل است.

PA-DSS و تاجران

تاجران (در واقع کارشناسان و مدیران IT و امنیت آنها) باید قادر به شناسایی آسیب‌پذیری‌های قانونی برنامه پرداخت باشند. آنها باید از توسعه‌دهندگان در مورد کنترل های امنیتی برنامه هایشان، فراتر از PCI سوال کنند. تاجران باید نقاط ضعف PA-DSS را یاد بگیرند تا اطمینان حاصل کنند که برنامه‌های آنها دارای مکانیزم‌های حفاظت کافی است و عدم امنیت محیط را جبران می‌کند.

PCI DSS

بر خلاف PA-DSS که در آن فرآیند دستیابی به وضعیت توافق صحت‌سنجی نامیده می‌شود، فرآیند PCI DSS مشابه، به عنوان ارزیابی نامیده می‌شود. تفاوت آن دو در اهداف‌شان است: برنامه تنها پرداخت‌کننده درPA-DSS، در مقابل محیط پردازش – پرداخت بی عیب درPCI DSS با وجودی که از لحاظ تکنیکی، فرآیند ارزیابی PCI DSS همانند ارزیابی PA-DSS است، تفاوت‌های قابل توجهی وجود دارد:

- **محدوده ارزیابی بسیار گسترده است.** (حتی از لحاظ فیزیکی - ممکن است شامل مکان‌های متعدد مانند فروشگاه‌های خرده فروشی، دفاتر، و مراکز داده باشد)، که منابع زیادی را از مامور بازرسی و تاجر مصرف می‌کند، و به آنها اجازه نمی‌دهد بر روی یک موضوع خاص متمرکز شوند.
- **PCI DSSیک استاندارد دوگانه است.** شکل و مقیاس ارزیابی به طور قابل ملاحظه‌ای وابسته به حجم تجارت تاجر یا ارائه‌دهنده خدمات تغییر می‌کند. تاجران براساس مقدار تراکنش‌های کارت که سالانه پردازش می‌کنند؛ به چندین سطح تقسیم می‌شوند. هر برند پرداخت برای سطح خودشان قواعد و نیازمندی‌های تعریف شده‌ای دارند؛ بنابراین قوانین مختلف از برندهای مختلف به منظور دریافت تصویری روشن از مسیر تصدیق باید باهم مرتبط باشند. تاجران، وابسته به سطح تجارت یا باید QSA را اجاره نمایند یا پرسش‌نامه خود ارزیابی یا (SAQ) را کامل کنند.
- **شورای PCI لیستی از سازمان‌های تأیید شده PCI DSS را نگه نمی‌دارد.**

نیازمندی‌های دوازده‌گانه و امنیت برنامه پرداخت

جدول ۳-۵ نشان می‌دهد که چگونه به کاربر (تاجر) نرم افزار (برنامه پرداخت) پیاده‌سازی کنترل‌های امنیتی تحمیل می‌شود که منطقا نباید بر عهده آنها باشد.

جدول ۳-۵: نیازمندی‌های PCI-DSS و نواحی آسیب‌پذیری کلیدی برنامه پرداخت

	نیازمندی PCI DSS	داده در حافظه	داده در موقع انتقال	داده در محل	تنظیمات و کد AP
۱	نصب و راه‌اندازی دیوارآتش برای محافظت از داده دارنده کارت	○	○	○	○
۲	عدم استفاده از پیش‌فرض‌های اعمال شده	-	○	○	-

	فروشنده برای رمزعبورهای سیستم و سایر پارامترهای امنیتی				
۳	محافظت از داده‌های دخیره شده دارنده کارت	-	-	•	-
۴	رمزنگاری انتقال داده‌های دارنده کارت در تقابل با شبکه‌های عمومی و باز	-	O	-	-
۵	استفاده و بروزرسانی منظم نرم‌افزار آنتی ویروس و برنامه‌ها	O	-	O	O
۶	توسعه ونگهداری برنامه‌ها و سیستم‌های امن	O	O	O	O
۷	محدود کردن دسترسی به داده‌های دارنده‌کارت با دانستن نیاز تجاری	-	-	-	-
۸	اختصاص ID منحصربفرد برای هر شخص با دسترسی رایانه‌ای	O	O	O	O
۹	محدود کردن دسترسی فیزیکی به داده‌های دارنده کارت	-	-	-	-
۱۰	نظارت و پیگیری همه دسترسی‌ها به منابع شبکه و داده‌های دارنده کارت	-	-	-	-
۱۱	تست منظم سیستم‌ها و پردازش‌های امنیتی	-	-	-	-
۱۲	برقراری سیاستی که امنیت اطلاعات را برای همه افراد نشان می‌دهد	-	-	-	-

توضیح اینکه:

•: اگرپیاده‌سازی به درستی باشد یک محافظت کامل و کافی ارائه می‌دهد.

O: حفاظت محدود شده و انتخابی را ارائه می‌دهد

- : هیچ‌گونه حفاظت مستقیمی را ارائه نمی‌دهد.

PCI DSS و تاجران کوچک

در جدول ۳-۶ نیازمندی‌های دوازده‌گانه PCI DSS و وابستگی سطح حفاظت به حجم تجارت نشان داده شده است.

جدول ۳-۶: کارایی نیازمندی‌های PCI-DSS وابسته به اندازه تاجر

	نیازمندی PCI DSS	تاجران بزرگ	تاجران کوچک
۱	نصب و راه‌اندازی دیوارآتش برای محافظت از داده دارنده کارت	•	-
۲	عدم استفاده از پیش‌فرض‌های اعمال شده فروشنده برای رمزعبورهای سیستم و سایر پارامترهای امنیتی	•	-
۳	محافظت از داده‌های دخیره شده دارنده کارت	○	○
۴	رمزنگاری انتقال داده‌های دارنده‌کارت در تقابل با شبکه‌های عمومی و باز	•	○
۵	استفاده و بروزرسانی منظم نرم‌افزار آنتی ویروس و برنامه‌ها	•	•
۶	توسعه ونگهداری برنامه‌ها و سیستم‌های امن	○	-
۷	محدود کردن دسترسی به داده‌های دارنده‌کارت با دانستن نیاز تجاری	•	-
۸	اختصاص ID منحصربفرد برای هر شخص با دسترسی رایانه‌ای	•	-
۹	محدود کردن دسترسی فیزیکی به داده‌های دارنده کارت	-	-
۱۰	نظارت و پیگیری همه دسترسی‌ها به منابع شبکه و داده‌های دارنده کارت	•	-
۱۱	تست منظم سیستم‌ها و پردازش‌های امنیتی	•	-
۱۲	برقراری سیاستی که امنیت اطلاعات را برای همه افراد نشان می‌دهد	•	-

توضیح اینکه:

•: می‌تواند توسط تاجر پیاده‌سازی شود.

○: تاجر به فروشنده برنامه پرداخت وابسته است.

- - واقع بینانه نیست که از تاجر انتظار یک پیاده‌سازی مناسب باشد. زیرا آن به تخصص‌های خاص و/یا منابع(گران) بیشتری نیاز دارد.

PCI DSS و توسعه‌دهندگان PA

بر خلاف PA-DSS که از برنامه پرداخت بطور ضمنی محافظت می‌کند؛ PCI DSS بر روی محیط عملیاتی کل، بیشتر از محصول‌های نرم‌افزاری خاص، متمرکز است. PA-DSS برای پشتیبانی از نیازمندی‌های PCI DSS پیچیده‌تر در نرم‌افزار پرداخت ایجاد شد. هیچ برنامه‌های پرداختی صحت‌سنجی PA-DSS را نمی‌گذرانند. قوانینی وجود دارند که محدوده قابلیت اجرایی، ارزیابی PA-DSS را تعیین می‌کنند. گروه‌های برنامه‌های پرداخت زیر برای ممیزی PA-DSS مطلوب نیستند اما توسعه‌دهندگان آنان هنوز باید از نیازمندی‌های PCI DSS یا PA-DSS پیروی کنند.

- برنامه‌های پرداخت درون سازمانی که توسط تاجران توسعه یافتند.
- برنامه‌های پرداخت که توسط فروشندگان نرم‌افزار برای مشتری‌های خاص توسعه یافتند.
- برنامه‌های پرداخت که قبلا تولید شده و به فروش می‌رسند؛ بدون نیاز به سفارشی‌سازی بیشتر بعد از نصب.

مقایسه نیازمندی‌های PA-DSS و PCI DSS

جدول ۳-۷ دو استاندارد را مقایسه می‌کند و نشان می‌دهد که همپوشانی نیازمندی‌ها کمی معنی‌دارتر است؛ بنابراین ممکن است برای متمایز کردن آنها از نیازمندی‌های مستقلی که از دیدگاه امنیت برنامه بلااستفاده‌اند؛ مفید باشد.

جدول ۳-۷: مقایسه نیازمندی‌های PA-DSS و PCI DSS

نیازمندی PA-DSS	نیازمندی PCI DSS	تضمین حفاظت قوی از:	تسهیل حفاظت اتفاقی از:
۱. عدم نگه‌داری نوار مغناطیسی کامل، کد یا مقدار تأییده کارت (CAV2، CID، CVC2، CVV2) یا داده‌های بلوکی PIN	۲,۳ عدم ذخیره‌سازی داده‌های احزار هویت شده حساس بعد از احراز هویت(حتی اگر رمزنگاری شده‌اند)	هیچ	داده در محل
۲. محافظت از داده‌های ذخیره شده دارنده‌کارت	۳. محافظت از داده‌های دخیره شده دارنده کارت	داده در محل	داده در محل
۳. فراهم کردن ویژگی‌های احراز هویت امن	۸. اختصاص ID منحصربفرد برای هر شخص با دسترسی رایانه‌ای	هیچ	داده در محل

۴. ثبت فعالیت برنامه پرداخت	۱۰. نظارت و پیگیری همه دسترسی‌ها به منابع شبکه و داده‌های دارنده کارت	هیچ	هیچ
۵. توسعه برنامه‌های پرداخت امن	۶. توسعه ونگهداری برنامه‌ها و سیستم‌های امن	هیچ	کد برنامه
۶. محافظت از انتقال‌های بی‌سیم	۴. رمزنگاری انتقال داده‌های دارنده‌کارت در تقابل با شبکه‌های عمومی و باز	هیچ	داده در موقع انتقال
۷ تست برنامه‌های پرداخت برای نشان دادن آسیب‌پذیری	۶. توسعه ونگهداری برنامه‌ها و سیستم‌های امن	هیچ	کد برنامه
۸ تسهیل کردن پیاده‌سازی شبکه امن		هیچ	هیچ
۹ داده‌های دارنده‌کارت نباید بر روی سرور متصل به اینترنت ذخیره شود	۷,۳,۱ قرار دادن اجزای سیستمی که داده‌های دارنده‌کارت را ذخیره می‌کنند(مانند پایگاه داده) در یک منطقه شبکه داخلی جدا شده از DMZ و دیگر شبکه‌های غیر قابل اعتماد	هیچ	داده در محل
۱۰ تسهیل کردن دستیابی راه دور امن برای برنامه پرداخت	۸,۳ ترکیب دو فاکتور احراز هویت برای دسترسی راه دور(دسترسی سطح شبکه ناشی از بیرون از شبکه) برای کارکنان مدیران و اشخاص ثالث	هیچ	هیچ
۱۱. رمزنگاری ترافیک حساس در کل شبکه‌های عمومی	۴. رمزنگاری انتقال داده‌های صاحبان کارت در تقابل با شبکه‌های عمومی و باز	هیچ	داده در موقع انتقال
۱۲. رمزنگاری تمام دسترسی‌های مدیریتی غیر کنسولی	۲,۳رمزنگاری تمام دسترسی های مدیریتی غیر-کنسول با استفاده از رمزنگاری قوی. استفاده از فناوری‌های مانندSSH، VPN ، SSL/TLS برای مدیریت های مبتنی بر وب و سایر دسترسی های مدیریتی غیر کنسولی	هیچ	داده موقع انتقال داده در محل
۱۳. حفظ اسناد آموزشی و برنامه‌های آموزشی برای مشتری‌ها، فروشندگان و افراد مرتبط با آنها		هیچ	هیچ
	۵ استفاده و بروزرسانی منظم نرم‌افزار آنتی ویروس و برنامه‌ها	هیچ	داده در محل، داده در حافظه، کدبرنامه
	۷. محدود کردن دسترسی به داده‌های دارنده‌کارت با دانستن نیاز تجاری	هیچ	هیچ

	۹. محدود کردن دسترسی فیزیکی به داده‌های دارنده کارت	هیچ	هیچ
	۱۱. تست منظم سیستم‌ها و پردازش‌های امنیتی	هیچ	هیچ
	۱۲. برقراری سیاستی که امنیت اطلاعات را برای همه افراد نشان می‌دهد	هیچ	هیچ

PTS

امنیت تراکنش PIN (PTS) برای فروشندگان برنامه پرداخت چندان مهم نیست. مگر اینکه برنامه آنها برروی صفحه رمز اجرا شود. در روش‌های POS یکپارچه، دستگاه‌های ورود PIN (PED) یا دستگاه‌های POI، که اشیا اصلی ارزیابی PTS هستند، با استفاده از رمزنگاری DES سه‌گانه و مکانیزم‌های مدیریت کلید DUKPT از امنیت PIN مراقبت می‌کنند. اگر از همان طرح برای محافظت داده کارت توسط پایانه‌های پرداخت استفاده شده بود؛ اکثریت مشکلات امنیت برنامه پرداخت را رفع می‌کرد.

P2PE

PCI P2PE اولین تلاش جدی است که واقعا سیستم منفرد امن ساخته شده از دستگاه‌های صفحه رمز تا POS، تا مرکز داده یک پایانه یا پردازنده پرداخت ایجاد می‌کند. علاوه بر این امنیت سیستم‌های P2PE طوری طراحی می‌شود که منحصرا به رمزنگاری قوی محافظت شده توسط سخت‌افزار به جای کنترل‌های زودگذر وابسته باشد. رمزنگاری مبتنی بر نرم‌افزار معمولا توسط فروشندگان نرم‌افزار پیشنهاد شد و توسط PCI DSS و PA – DSS کنار نهاده شد.P2PE چرخه فرآیند پرداخت کاملی را پوشش می‌دهد. همچنین ساختاری استاندارد که شامل ۶ حوزه است، تعیین می‌کند:

- حوزه۱: مدیریت دستگاه رمزنگاری
- حوزه ۲: امنیت برنامه
- حوزه ۳: محیط رمزنگاری
- حوزه ۴: قطعه‌بندی محیط‌های رمزگشایی و رمز نگاری
- حوزه ۵: محیط رمز گشایی و مدیریت دستگاه
- حوزه ۶: عملیات کلید رمزنگاری P2PE

دستور العمل PCI

پیروی از PCI DSS و PA-DSS طبق قانون نیاز نیست .با این حال، اگر شما برنامه پایانه فروشی را توسعه داده‌اید و می‌خواهید آن را به فروش برسانید، لازم است که صحت‌سنجی PA-DSS را به منظور به دست آوردن برخی از مزایایی مانند توانایی فروش محصول خود به هر مشتری، بگذرانید.

اشتباه جایگزینی با نشانه‌ها

نشانه‌گذاری برای کاهش مسؤلیت تاجران از پذیرش PCI DSS با جایگزینی شماره حساب اصلی (PAN) کارت اعتباری اصلی با یک جایگرین بنام نشانه[۳۳] یا Token به صنعت کارت اعتباری معرفی شده بود. نشانه بصورت منحصر بفرد شماره حساب اصلی را شناسایی می‌کند و آن را در پایگاه داده پرس‌وجو ارائه می‌دهد. هرچند، امنیت شماره‌های حساب اصلی تحت تاثیر قرار نمی‌گیرد. فناوری‌های نشانه‌گذاری مختلفی وجود دارند که برای تولید نشانه از روش‌های متفاوتی مانند توابع درهم‌ساز، تولید تصادفی شناسای منحصربفرد سراسری[۳۴]، رمزنگاری الگوهایی با تقلید از قالب اصلی شماره‌های حساب کارت استفاده می‌کنند. نشانه‌ها می‌توانند در پایانه فروش توسط برنامه پرداخت مشتری یا در مرکز داده توسط سرور برنامه پرداخت یا پردازنده پرداخت تولید شوند. جدول ۳-۸ نشان می‌دهد که چگونه پیاده‌سازی روش‌های نشانه‌گذاری، ناحیه‌های متفاوتی از آسیب‌پذیری‌های برنامه پرداخت را تحت تأثیر قرار می‌دهد.

جدول ۳-۸: نشانه‌گذاری و آسیب‌پذیری‌های برنامه پرداخت

ناحیه آسیب‌پذیری برنامه پرداخت	محافظت ارائه شده توسط نشانه‌گذاری به PA در POS
حافظه	–
حافظه موقت (S&F، TOR، تراکنش‌های فعال)	–
حافظه بلند مدت (دسته، رکوردهای مستقر)	○
حافظه بلند مدت (بایگانی های تراکنش)	•
ثبت کردن فایل	○
ارتباط محلی	-
ارتباط بین دستگاه POI و POS	-
پیوند به پردازنده‌ها	-
تنظیمات و کد برنامه	-

توضیح اینکه:

[۳۳] Token

[۳۴] Globally Unique Identifier(GUID)

•: اگرپیاده‌سازی به درستی باشد یک محافظت کامل و کافی ارائه می‌دهد.

O: حفاظت محدود شده و انتخابی را ارائه می‌دهد

- : هیچ‌گونه حفاظت مستقیمی را ارائه نمی‌دهد.

خلاصه فصل

PA-DSS و PCI DSS حتی اگر بطور کامل پیاده‌سازی شوند، حداقل نیازمندی‌ها را فراهم کرده و هیچ محافظتی در برابر تهدیدها در سه مورد از چهار مورد آسیب‌پذیری‌های کلیدی برنامه پرداخت: داده در حافظه، داده در موقع انتقال، تنظیمات و کد برنامه ندارد. هر دو PA-DSS و PCI DSS محافظت قابل ملاحظه (اما نه کامل) در یکی از چهار نواحی آسیب‌پذیری کلیدی – داده ذخیره‌شده- تسهیل می‌کنند. در صورتی که اگر فروشنده برنامه مکانیزم‌های رمزنگاری قوی را پیاده‌سازی کند.

PCI DSS محافظت کافی و مناسبی برای سیستم‌های پرداخت و برنامه‌های پرداخت مربوط به تاجران کوچک فراهم نمی‌کند.

بسیاری از نیازمندی‌های PA-DSS و PCI DSS برای سازمان‌های بزرگ، محیط‌های مرکز داده یا برنامه‌های وب طراحی شده‌اند؛ بنابراین زمانی که به سیستم‌های پایانه فروش معمول اعمال می‌شوند، مؤثر نیستند.

نشانه‌گذاری برای داده‌های حساس محافظت کافی و مناسب فراهم نمی‌کند؛ زیرا بر روی ناحیه منفرد از آسیب‌پذیری برنامه پرداخت متمرکز می‌شود.

فصل چهارم:

سرقت داده کارت

در این فصل در مورد امنیت برنامه پرداخت و به خصوص نحوه محافظت از آن در برابر سرقت بحث خواهد شد. در این زمینه سوالاتی مطرح است، از جمله اینکه صاحبان کارت دارای چه نوع داده‌هایی می‌باشند و چرا باید از آنها محافظت نماییم؟ چه نوع اطلاعات به سرقت می‌روند؟ در صورتی که دزدیده شوند، آیا کسب درآمد از آنها امکان پذیر است؟ و سوالات دیگری از این جنس که در این فصل درباره آنها بحث خواهد شد.

کارت جادویی

بسیاری از ما با اکثر کارت‌های پرداخت که شامل کارت اعتباری، کارت خدمات بانکی عادی و کارت هدیه می‌شود آشنا هستیم. تفاوت اساسی مابین کارت‌های مذکور این است که در خدمات کارت اعتباری عملیات مالی از پولی که به صادرکنندگان کارت بدهکاریم، در کارت بانکی عادی از پولی که در حساب خود داریم و در کارت هدیه با پولی که قبلا خریده‌ایم انجام می‌شود. از دید امنیتی، کارت بانکی عادی امن‌ترین نوع کارت است چون مستلزم کد شناسایی شخصی بوده و برای انجام عملیات پرداخت صحت‌سنجی می‌شود. البته کارت‌های بانکی دومنظوره‌ای نیز وجود دارد که بدون کد شناسایی شخصی هم مورد استفاده قرار می‌گیرند.

ساختار فیزیکی و ویژگی‌های امنیتی

قبل از ادامه بحث، ابتدا مروری بر ساختار فیزیکی کارت‌های پرداخت داشته باشیم که شاید بدیهی به نظر برسد. چون این ساختار نیز بخشی از امنیت اطلاعات بوده و امنیت پرداخت‌های الکترونیکی هیچ استثنایی ندارد. در واقع داده‌ها بدون ساخت نسخه کپی دقیق از کارت غیرقابل استفاده می‌باشد. برای تشخیص کارت پرداخت واقعی با جعلی چندین نشانه وجود دارد:

- **تصویر، رنگ زمینه و لوگوی برند بانک** برای تشخیص برندها و انواع مختلف کارت استفاده می‌شود و به همین دلیل نمی‌توان به آن به عنوان کنترل امنیتی قوی اعتماد کرد. طرح ظاهری کارت را می‌توان به راحتی توسط چاپگر PVC می‌توان جعل کرد. (چون استانداردی برای طراحی ظاهری وجود ندارد). اکثر بانک‌های صادر کننده کارت نیز دارای کارت‌هایی با طراحی متفاوتی هستند. و به دلیل عدم وجود استاندارد سازی در این زمینه، تقریبا شناسایی کارت اصلی از جعلی تنها با نگاه کردن به آن امری امکان ناپذیر به نظر می‌رسد.
- **شماره حساب اصلی[۳۵]، تاریخ انقضا، و نام صاحب حساب برجسته** (شکل ۴-۱) از زمان کارت اعتباری دستی وجود داشته است و تنها راه انجام فرایند تراکنش بوده است. امروزه از داده‌های برجسته به عنوان یک ویژگی امنیتی اضافی کاربرد دارد هرچند که قدرتمند نیست و یک دستگاه چاپ حروف برجسته را می‌توان به قیمت ۳۰۰ دلار خریداری نمود.

[۳۵] Primary Account Number (PAN)

- **CVV2**[۳۶]، یک کد سه تا چهار رقمی است که بر روی کارت حک شده است (در شکل ۴-۲ با دایره علامتگذاری شده). از آن عمدتا در تراکنش‌های برخط که بدون حضور کارت است استفاده می‌شود و به ندرت در تراکنش‌هایی که مستلزم استفاده فیزکی کارت است به آن نیاز خواهیم داشت.
- **علامت‌های فرابنفش**، که بر روی کارت قرار گرفته است (شکل ۴-۳)، و توسط نورهای فرابنفش می‌توان آن‌را خواند. حسابدارها مانند چک کردن گواهی‌نامه رانندگان می‌توانند این علامت فرابنفش را چک کنند. هرچندکه، هیچ‌کس بدلیل افزایش زمان پرداخت، این‌کار را انجام نمی‌دهد. و این علامت فرابنفش را می‌توان با پرینتر حاوی جوهر فرابنفش جعل کرد.

شکل ۴-۱: قسمت جلویی کارت پرداخت

شکل ۴-۱: قسمت پشتی کارت پرداخت

شکل ۴-۲: کارت اعتباری زیر نور عادی

[۳۶] Card Verification Value (CVV2)

شکل ۴-۳: کارت اعتباری زیر نور سیاه که علامت‌های فرابنفش آشکار می‌شود

- **شماره تماس واحد خدمات مشتری** که امکان تماس با نماینده ارائه خدمات بانک صادرکننده کارت را جهت جعلی نبودن کارت و همچنین وجود اعتبار کافی جهت انجام تراکنش را میسر می‌سازد. البته این کار زمانی رخ می‌دهد که مبلغ تراکنش بسیار بالا بوده یا سیستم اصالت سنج بصورت خودکار تراکنش را لغو کند.
- **ظاهر فلزی** (اختیاری) که از روکش نقره‌ای یا طلا بر روی اعداد و نوشته‌های برجسته روی کارت استفاده می‌شود تا ظاهری بهتر داشته باشد. و این ویژگی توسط دستگاه مهرزن ورقه داغ براحتی جعل می‌شود.
- **امضای صاحب کارت** (اختیاری) در پشت کارت (شکل ۴-۲) که برای مقایسه امضای صاحب کارت به هنگام تراکنش است که معمولا هیچ مشتری این کار را انجام نمی‌دهد.
- **عکس صاحب کارت** (اختیاری)، که به پشت یا جلوی کارت اضافه می‌شود که کارت را به شناسه تصویری تبدیل می‌کند.
- **هولوگرام** (اختیاری) با علامت برند بانک صادرکننده در جلو یا پشت کارت چاپ می‌شود.
- **نوار مغناطیسی هولوگرافی** (اختیاری) که یک ویژگی جدید به شمار می‌آید و به آسانی نمی‌توان آن‌را جعل کرد.

علت شکست ویژگی‌های امنیتی

چندین مسئله در رابطه با کنترل امنیت فیزیکی کارت‌های پرداخت وجود دارد:

- **مکانیزم‌های حفاظتی مداوم نیست.** چندین استاندارد غیررسمی مانند استفاده از شماره حساب برجسته شده، تاریخ انقضا، و نام صاحب کارت وجود دارد اما هنوز استاندارد رسمی موجود نیست. برای مثال شماره تایید کارت، چندین نام مختلف (CSC, CVV, CAV, CVC, CID, CAV2, CVC, CVV2)، چندین طول مختلف (سه یا چهار رقم) و چندین موقعیت مختلف (مسیرهای مغناطیسی در جلو یا پشت کارت) دارد که یک حسابدار تازه کار را گیج می‌کند.

- **بسیاری از ویژگی‌های امنیتی اجباری نیست.** برای مثال، چاپ عکس صاحب کارت بر روی کارت یک ایده بسیار خوبی برای جلوگیری از کلاهبرداری است یا نوارهای مغناطیسی هولوگرافی یک مثالی دیگری است تا یک محافظت قدرتمند فیزیکی را فراهم کند و چون اجباری نبوده و مستلزم استاندارد خاصی نیست پس کاربرد گسترده‌ای ندارد.
- **کارت‌های پرداخت، کنترل فیزیکی ندارد تا بتواند توسط نوار مغناطیس‌خوان تاییدیه خودکار را انجام بدهد.** همه حفاظت‌های فیزیکی برای تایید انسانی طراحی شده‌اند. بنابراین، پایانه‌های پرداخت بدون تاییدیه انسان، مانند ATM، جایگاه گاز یا کیوسک‌های خودکار، چون شخصی برای صحت‌سنجی کارت حضور ندارد پس ساده‌ترین هدف‌های کلاهبرداری هستند.
- **بیشتر کنترل‌های فیزیکی براحتی می‌تواند جعل شود.** برخلاف وجود تکنولوژی پیشرفته مانند چیپ‌های EMV، کنترل‌های فیزیکی عادی براحتی در خانه می‌تواند جعل شود.
- **تایید ویژگی‌های حفاظت فیزیکی بالاجبار نبوده و اغلب توسط فروشندگان حذف می‌گردد.** هرچند کارت‌های اعتباری کنترل‌های فیزیکی نسبتا قوی مانند نوارهای مغناطیسی هولوگرافی دارند، اما حسابدارها به‌ندرت از آن استفاده می‌کند. بسیاری از تاجران حتی از این کنترل فیزیکی استفاده نمی‌کنند. چون چنین کنترل‌هایی جذابیت فرآیند پرداخت را کاهش داده و زمان فرآیند تراکنش را افزایش می‌دهد که مشخص کننده تعداد مشتریان یک حساب‌دار در شیفت کاری است، و منجر به استخدام حسابدار و نگهبان زیاده بوده که مقرون به‌صرفه نیست.

بنابراین، همه روش‌های محافظتی غیر خودکار کارت‌های پرداخت مستلزم تاییدیه دستی بوده و بنابراین بیشترشان غیرفعال است، پس راه را برای کلاهبرداری باز می‌کند.

بخش درونی نوار مغناطیسی

اطلاعات صاحب کارت در پشت کارت پرداخت و در داخل نوار مغناطیسی قرار دارد. دو مسیر (مسیر۱ و مسیر۲) برای فرایند پرداخت‌های الکترونیکی وجود دارد. البته مسیر سومی با نام مسیر۳ موجود است که در تراکنش‌های پایانه فروش استفاده نمی‌شود. فرمت داده مسیرهای مغناطیسی توسط استاندارد ISO7813 تعریف می‌شود و برای همه کارت‌های پرداخت یکسان است. وقتی کارت مغناطیسی در فروشگاه کشیده می‌شود، دستگاه نوار مغناطیس‌خوان مسیرهای نوار مغناطیسی را می‌خواند و به پردازشگر پرداخت ارسال می‌کند و به سپس به چندین زیرشاخه از جمله تاریخ انقضای شماره حساب اصلی، کد سرویس، و داده‌های دیگر تجزیه می‌شود.

نوارهای مغناطیسی روش تقریبا موثری است اما از نظر امنیت، تکنولوژی هوشمندی نیست. نوار مغناطیسی می‌تواند کپی شده و توسط یک رمزکننده[۳۷] نوار مغناطیسی در یک کارت خالی نوشته شود. مسیر۱ و ۲ مولفه‌های بسیاری مهمی در فرآیند پرداخت هستند. هر دو مسیر می‌تواند برای پرداخت تمام فیزیکی در هر تراکنشی در داخل فروشگاه و برداشت‌های ATM استفاده شود. شماره حساب اصلی اجازه خریدهای برخط را می‌دهد که در هر دو مسیر موجود است.

مسیر۱

این مسیر شامل تمام اطلاعات صاحب کارت از قبیل نام و نام خانوادگی است. بیشترین طول مسیر۱ برابر ۱ تا ۷۹ بایت است. اولین بایت شماره حساب اصلی، سپس نام و نام خانوادگی صاحب کارت و سپس تاریخ انقضای کارت و سپس کد سرویس که با کاراکتر "^" از هم جدا شده است و در جدول ۴-۱ برای مثال زیر مورد استفاده قرار گرفته است:

%B4005554444444403^GOMZIN/SILVA^1521101000000012300?

جدول ۴-۱: مثالی از مولفه‌های مسیر۱

بخش	تاریخ
شماره حساب اصلی	4005554444444403
تاریخ انقضا	1512 (دسامبر 2015)
نام صاحب کارت	SILVA
نام خانوادگی صاحب کارت	GOMZIN
کد سرویس	101
CVV	123

هر دو مسیرهای ۱ و ۲ حفاظت نشده‌اند و داده در یک متن ساده ذخیره می‌شود. به‌عبارت دیگر، در صورت بدست آوردن ۷۹ بایت از مسیر۱ یا فقط ۴۰ بایت از مسیر۲ می‌توانید یک کپی کامل از کارت پرداخت را داشته باشید. مسیر۱ فقط شامل کاراکترهای عدد و الفبایی است. جدول ۴-۲ ساختار مسیر۱ را با جزئیات بیشتری بررسی می‌کند.

[۳۷] Encoder

جدول۴-۲: ساختار مسیر۱ با جزئیات بیشتر

مولفه	طول (بایت)	نوع داده و فرمت	توضیحات
%	۱	همیشه کاراکتر "%"	شروع کدگذاری
B	۱	همیشه کاراکتر "B"	کد فرمت
شماره حساب اصلی	حداکثر ۱۹	ارقام ۰ – ۹	رجوع به توضیحات بخش شماره حساب اصلی
^	۱	همیشه کاراکتر "^"	جداساز مابین شماره حساب اصلی و نام صاحب حساب
نام	۲ – ۲۶	کاراکتر	نام خانوادگی
			جداساز "/"
			نام
^	۱	همیشه کاراکتر "^"	جداساز مابین نام صاحب حساب و تاریخ انقضا
تاریخ انقضا	۴	ارقام	معمولا سال ماه
کد سرویس	۳	ارقام	شامل سه زیر بخش است
داده اختیاری	متغیر (اما نباید طول مسیر از ۷۹ بایت فراتر رود)	کاراکتر	اطلاعات اختیاری مربوط به نوع کارت.
؟	۱	همیشه کاراکتر "؟"	پایان کدگذاری
LRC	۱	رقم	فراوانی طولی تعریف شده در ISO7811-2. ، معمولا توسط نوار مغناطیس‌خوان تایید می‌شود تا صحت داده

خوانده شده را بررسی کند.

مسیر۲

این مسیر شامل اطلاعات خلاصه برای فرایند پرداخت است: شماره حساب اصلی، تاریخ انقضا، کد سرویس و اطلاعات اضافی که به صادرکننده و نوع کارت مربوط است. مسیر۲ شکل خلاصه‌تر مسیر۱ است تا ترمینال‌هایی که از خطوط زمینی تلفن استفاده می‌کنند بتوانند با تاییدکنندگان، ارتباط موثرتری برقرار کنند.

مسیر۲ با اکثر گیرندگان پرداخت سازگار است، بنابراین مسیر۱ در بیشتر برنامه‌ها اختیاری خواهد بود. طول استاندارد مسیر۲ برابر ۴۰ کاراکتر اسکی خواهد بود که هر اسکی طولی برابر ۱ بایت دارد. ارقام اول (معمولا ۱۶ است اما از ۱۵ تا ۱۹ متفاوت است) برای شماره حساب اصلی رزرو شده است. کاراکتر "=" شماره حساب اصلی را از ۴ بایت برای تاریخ انقضا و اطلاعات اختیاری مجزا می‌کند. نمونه‌ای از مسیر۲ در جدول ۴-۳ برای مثال زیر آمده است.

```
;4005554444444403=1512101000000012300?
```

جدول۴-۳: مولفه‌های نمونه مسیر۲

بخش	داده
شماره حساب اصلی	4005554444444403
تاریخ انقضا	1512
کد سرویس	101
CVV	123

جدول ۴-۴ ساختار مسیر۲ را بصورت جزئی‌تر بررسی می‌کند.

جدول۴-۴: ساختار جزئی‌تر مسیر۲

جدول ۴ مولفه	طول (بایت)	نوع داده و فرمت	توضیحات
;	۱	همیشه کاراکتر ";"	شروع کدگذاری
شماره حساب اصلی	حداکثر ۱۹	ارقام ۰ – ۹	مشابه مسیر۱
=	۱	همیشه کاراکتر "="	جداساز مابین شماره حساب اصلی و تاریخ

			انقضا
تاریخ انقضا	۴	ارقام	مشابه مسیر۱
کد سرویس	۳	ارقام	مشابه مسیر۱
اطلاعات اضافی	متغیر (اما نباید طول کلی از ۴۰ بایت فراتر رود)	ارقام	مشابه مسیر۱
؟	۱	همیشه کاراکتر "؟"	پایان کدگذاری
LRC	۱	رقم	مشابه مسیر۱

شماره حساب اصلی

این شماره همراه با چند استثنا (American Express که ۱۵ رقم دارد) معمولا ۱۶ رقم دارد. بیشترین طول شماره حساب اصلی برای کارت‌هایی مانند کارت سوخت ۱۹ رقم است.

شماره حساب اصلی در روی کارت های پرداخت برجسته شده است که در صورت خراب بودن نوار مغناطیس‌خوان یا خراب بودن سیستم الکترونیکی، به صاحب حساب اجازه وارد کردن دستی آن و انجام عملیات پرداخت را در پایانه فروش می‌دهد که البته این روش آسان‌ترین و قدیمی‌ترین روش سرقت کردن پول صاحب‌حساب بدون هک کامپیوتر است.

جدول ۴-۵ موقعیت شماره حساب اصلی را در مسیرهای مغناطیسی نشان می‌دهد.

جدول۴-۵: موقعیت شماره حساب اصلی در مسیرهای مغناطیسی

مسیر	داده
۱	%B**4005554444444403**^GOMZIN/SILVA^1521101000000012300?
۲	**;4005554444444403**=1521101000000012300?

هر دو مسیر ۱و ۲ شامل شماره حساب اصلی یکسانی هست، که از چندین مولفه پیشوند استاندارد، شماره حساب، و رقم کنترلی تشکیل شده است. داشتن کپی کامل از مسیرهای ۱و ۲ برای جعل کافی است اما در برخی مواقع مانند خریدهای برخط، دانستن تنها شماره حساب اصلی برای جعل خرید کافی می‌باشد.

تاریخ انقضا

تاریخ انقضا با چهار رقم، دو رقم برای ماه و دو رقم برای سال کد شده است. هر دو مسیر شامل تاریخ انقضا هستند که بعد از کاراکتر جداکننده "=" در مسیر ۱ و بعد از دومین جداکننده "^" در مسیر ۲ قرار می‌گیرند. (جدول ۴-۶). ترتیب ماه و سال توسط استاندارد ISO تعیین شده است. به این صورت که دو رقم اول نشان‌دهنده سال و دو رقم بعدی نشان‌دهنده ماه می‌باشند. برای مثال تاریخ انقضای کارت در صورتی که دسامبر ۲۰۱۵ باشد، به صورت 1512 کدگذاری می‌شود.

جدول ۴-۶: موقعیت تاریخ انقضا در مسیرهای مغناطیسی

مسیر	داده
۱	%B4005554444444403^GOMZIN/SILVA^**1521**101000000012300?
۲	;4005554444444403=**1512**101000000012300?

برنامه‌های پرداخت و بیشتر پردازش کنندگان پرداخت و دریافت کنندگان کارت، تاریخ انقضا را کنترل نمی‌کنند. بنابراین هر عدد تصادقی را می‌توان بجای این تاریخ بر روی کارت نوشت چون صحت آن در پایگاه داده بررسی نمی‌شود.

تاریخ انقضا چه در هنگام انتقال و چه در زمان استقرار در محل ذخیره سازی معمولا توسط برنامه‌های پرداخت رمزگذاری نمی‌شود (استانداردهای PCI آن را لازم نمی‌دانند) و براحتی می‌توان آنها را از فایل‌های ثبت شده، داده‌های حافظه، استراق سمع ارتباطات و پایگاه داده بدست آورد.

محدوده شماره حساب بانک[۳۸] و پیشوند ISO

۶ رقم اول شماره حساب اصلی (جدول ۴-۷) پیشوند ISO نامیده می‌شود، که با نام‌های BIN, BIN Prefix هم بیان می‌شود.

جدول ۴-۷: موقعیت پیشوند ISO در مسیرهای مغناطیسی

مسیر	داده
۱	%B**400555**4444444403^GOMZIN/SILVA^1521101000000012300?

[۳۸] Bank Identification Number (BIN)

;4005554444444403=15121010000000012300? ۲

از لحاظ فنی پیشوند ISO بخشی از شماره حساب نیست چون شامل اطلاعات صادرکننده کارت می‌باشد، بنابراین شماره حساب واقعی که صاحب کارت را شناسایی می‌کند تنها شامل ۹ رقم است. یک شماره حساب ۱۶ رقمی شبیه زیر است:

۹ رقم = ۱ رقم کنترل – ۶ رقم پیشوندISO – ۱۶ رقم شماره حساب PAN

چند دلیل برای افشای ۶ رقم اول شماره حساب اصلی توسط استانداردهای PCI وجود دارد:

- پیشوند ISO شماره مخفی نیست چون صاحبان کارت را شناسایی نمی‌کند و بسیاری از صاحبان کارت دارای پیشوند ISO یکسانی هستند
- پیشوند ISO را می‌توان بصورت مجازی ایجاد کرد چون حاوی اطلاعات عمومی است.
- بیشتر سرویس‌های پرداخت مشتری به منظور هدایت درست تراکنش به پردازشگر متناسب باید پیشوند ISO را بدانند. به همین دلیل با اینکه داده‌های حساس کارت توسط رمزگذاری نقطه به نقطه رمزگذاری می‌شوند ولی سیستم همچنان باید قادر به مشاهده پیشوند ISO برای پردازش تراکنش‌ها باشد.

در شماره حساب اصلی ۱۶ رقمی، از آنجا که ۶ رقم اول و ۴ رقم آخر تقریبا همیشه به صورت رمز نشده در دسترس هستند و رقم کنترل تابعی است که بر اساس مقدار شماره حساب اصلی محاسبه می‌شود، دانستن این حقیقت برای شکستن الگوریتم تابع درهم‌ساز بسیار مهم است چون فقط ۶ رقم باقیمانده بایستی بدست بیاید.

اولین رقم در پیشوند ISO، MII [۳۹] نامیده می‌شود و برای شناسایی صنعت به کار می‌رود. برندهای پرداخت کلان مقدار MII از ۱ تا ۶ به خود اختصاص داده‌اند و با استفاده از چهار رقم اول پیشوند ISO می‌توانند شناسایی شوند.

جدول ۴-۸: محدوده شماره شناسایی بانک

MII	محدوده BIN	برند پرداخت
۱	1800xx	JCB
2	2131x	JCB
3	34xxxx, 37xxxx	American Express
	300xxx – 305xxxx, 38xxxxx	Diners Club

[۳۹] Major Industery Identifier

JCB	35xxxx	
Visa	4xxxxx	4
MasterCard	51xxxx – 55xxxx	5
Discover	6011xx. 65xxxx	6

رقم کنترل شماره حساب اصلی

آخرین رقم هر شماره حساب اصلی رقم کنترل یا بررسی نامیده می‌شود که در هنگام جستجو برای شماره حساب اصلی در حافظه یا فایل، تعیین کننده تفاوت مابین حساب واقعی با دنباله‌ای از ارقام تصادفی است. رقم بررسی در واقع بخشی از شماره حساب نیست. بطور مثال اگر طول شماره حساب ۱۶ رقم باشد در این صورت ۱۵ رقم جای می‌گیرد و رقم ۱۶ آن همان رقم بررسی خواهد بود (جدول ۴-۹). رقم بررسی توسط محاسبه ماژوله ۱۰ (Mod 10) مشخص می‌شود، و به فرمول لون[۴۰] معروف است.

جدول ۴-۹: موقعیت رقم بررسی شماره حساب اصلی در مسیرهای مغناطیسی

مسیر	داده
۱	%B4005554444444403^GOMZIN/SILVA^15211010000000012300?
۲	;4005554444444403=15211010000000012300?

اعتبار سنجی رقم بررسی در مسیرها لازم نیست چون توسط جداکننده‌های داده و با الگوهای شماره حساب اصلی و تاریخ انقضا کاملا مشخص است.

کد سرویس

سه بایت کد سرویس (جدول ۴-۱۰) شامل زیر مجموعه ۳۱بایتی است. مقادیر این زیرمجموعه‌ها به دلیل تعریف چگونگی کنترل کارت توسط برنامه پرداخت و پردازنده پرداخت در حین تایید پرداخت بسیار مهم است.

جدول ۴-۱۰: موقعیت کد سرویس در مسیرهای مغناطیسی

مسیر	داده
۱	%B4005554444444403^GOMZIN/SILVA^15211010000000012300?

[۴۰] Luhn

متداولترین کده‌های سرویس، ۱۰۱ و ۲۰۱ است. اولین رقم مشخص کننده وجود یا عدم وجود چیپ EMV است. بنابراین، کارت‌های شامل ۱۰۱ نوار ساده مغناطیسی است که بطور مجازی در هر جای کشور مورد استفاده قرار می‌گیرد، درحالی‌که کارت‌های ۲۰۱ شامل چیپ بوده و محدودیت‌هایی را در استفاده دارد.

جدول ۴-۱۱ دستورالعمل‌های مهم اضافی که توسط کدهای سرویس تعریف شده است را برای برنامه‌های پرداخت و پردازشگرها نشان می‌دهد. این دستورالعمل‌ها می‌تواند شامل ناحیه پذیرش، محل شماره شناسه شخصی، و محدودیت‌های تولی باشد.

جدول ۴-۱۱: دستورالعمل کد سرویس

دستورالعمل		کد سرویس		
		اولین رقم	دومین رقم	سومین رقم
تکنولوژی	کارت شامل چیپ است	۲ یا ۶		
پذیرش	بین‌المللی	۱ یا ۲		
	داخلی	۵ یا ۶		
احراز هویت	معمولی		۰	
	توسط کاربر		۲	
محدودیت‌ها	ندارد	۰، ۱ یا ۶		
	کالاها و خدمات			۲، ۵، یا ۷
	فقط ATM			۳
ملزومات شناسه شخصی	شماره نیازمند شناسایی شخصی			۰، ۳، یا ۵

مقادیر تایید کارت

مقادیر تایید کارت یا Card Verificaton Value توسط برندهای کارت ایجاد شده تا علیه کلاهبرداری‌های کارت‌های اعتباری مقابله شود. هر برند پرداخت، قرارداد نامگذاری مختص خود را دارد که یکی دیگر از نتایج عدم استاندار سازی در صنعت است. و به همین‌دلیل دو نوع مقدار تایید کارت وجود دارد:

اولین گروه کدها (CVV) نامیده می‌شوند که در جدول ۴-۱۲ لیست شده و بصورت اطلاعات اختیاری در در داخل مسیرهای مغناطیسی ۱ و ۲ قرار داده شده است.

جدول ۴-۱۲: مقادیر تایید کارت که بر روی نوارهای مغناطیسی کد شده است

برند پرداخت	کد	نام	موقعیت	طول
American Express	CSC	کد امنیت کارت	مسیر ۱ و ۲	۳
Discover	CVV	مقدار تایید کارت	مسیر ۱ و ۲	۳
JCB	CAV	مقدار صحت کارت	مسیر ۱ و ۲	۳
MasterCard	CVC	کد اعتبار سنجی کارت	مسیر ۱ و ۲	۳
Visa	CVV	کد تایید کارت	مسیر ۱ و ۲	۳

دومین گروه از کدها CVV2 هستند که مانند CVV بوده و هر برند پرداختی نام‌های مختص خود را دارد (جدول ۴-۱۳).

جدول ۴-۱۳: مقادیر تایید کارت که بر روی قاب کارت چاپ شده است

برند پرداخت	کد	نام	موقعیت	طول
American Express	CID	شماره شناسایی کارت	جلوی کارت	۴
Discover	CID	شماره شناسایی کارت	پشت کارت	۳
JCB	CAV2	مقدار صحت‌سنجی کارت ۲	پشت کارت	۳
MasterCard	CVC2	کد تایید کارت ۲	پشت کارت	۳

Visa	CVV2	کد تایید کارت ۲	پشت کارت	۳

CVV کدگذاری شده در مسیرهای مغناطیسی

CVV که در ناحیه اختیاری در مسیرهای مغناطیسی ۱و ۲ کدگذاری شده‌اند در جدول ۴-۱۴ نشان داده شده است و برگرفته از مولفه‌های مسیر داده و با استفاده از تابع رمزنگاری خاص است.

جدول ۴-۱۴: موقعیت مقدار تایید کارت در مسیرهای مغناطیسی

مسیر	داده
۱	%B4005554444444403^GOMZIN/SILVA^1521101000000012300?
۲	;4005554444444403=15121010000000123300?

مقدار تایید کارت توسط گیرندگان میزبان و در حین فرآیند احراز هویت و هنگامی‌که پایانه فروش تمام داده‌های مسیرها را ارسال می‌کند، تطبیق داده می‌شود. قبلا، در کلاهبرداری‌های کارت اعتباری، سارقان کار آسانی داشته و اجباری در به دست آوردن کل مسیر برای ایجاد کارت تقلبی نداشتند. با مرور زمان شماره حساب اصلی و تاریخ انقضا به اندازه کافی شامل اطلاعات بود که بتوان تمام مسیر۲ را بدست آورد.

امروزه به لطف CVV، مسیرها را نمی‌توان براحتی بازسازی کرد و سارقان یا باید تمام مسیر را بدست آورند و یا ارقام آن را تخمین بزنند که با ۱۰۰۰ مقدار احتمالی مشکل است. مسیرهای ۱ و ۲ بازسازی شده از شماره حساب اصلی را هنوز می‌توان برای تراکنش‌های برون‌خطی استفاده کرد چون برنامه‌های پرداخت قابلیت صحت‌سنجی CVV به صورت برون‌خط را ندارند.

CVV2 چاپ شده روی کارت

اعتبار سنجی CVV2 اختیاری است و بعنوان یک سرویس اضافی می‌تواند برای خرید و فروش استفاده شود. این کد در فروشگاه‌های کوچک به دلیل حضور فیزیکی و استفاده توسط کارپرداز به ندرت اعتبارسنجی می‌شود. این کد زمانی مفید است که کارت حضور فیزیکی نداشت باشد. اگر داده کارت در پرداخت برخط از سیستم قربانی ربوده شود، می‌توان با گرفتن کپی از آن اطلاعات CVV2 را بدست آورد.

عبارات منظم[۴۱]

عبارات منظم (regex) تکنیکی از برنامه نویسی است که بدافزارها با استفاده از آن می‌توانند مولفه‌های شماره حساب اصلی و مسیر۱ و ۲ را در داخل حافظه کامپیوتر یا فایل‌های دیسک پیدا کنند. عبارات منظم نمادی برای مشخص سازی الگوی متن بجای بیان کاراکترهای رشته‌ای دقیق آن است. کارت‌های اعتباری و بانکی الگوهای مشخصی داشته و به‌راحتی می‌توان به دستورالعمل‌های عبارت منظم تبدیل کرد. این دستورالعمل‌ها را می‌توان توسط نرم افزار پیاده سازی کرد تا بتوان مسیرهای واقعی و شماره حساب‌های بدست آمده از حافظه یا فایل‌ها را مسدود کرد. برای مثال، همه شماره حساب‌های MasterCard با شماره ۵۱ تا ۵۵ شروع می‌شود، و ۱۴ رقم باقیمانده نیز هر رقمی مابین ۰ تا ۹ می‌تواند باشد، برای مثال ۵۱۰۰۰۰۰۰۰۰۰۰۰۰۰. پس عبارت منظم برای این مثال $[14][9-1][5-1]5^ خواهد بود.

اگر از چنین الگوهای ساده‌ای استفاده نماییم، به‌هنگام جستجو در داخل سیستم ممکن است با false positive بسیاری مواجه شویم یعنی شماره‌های تصادفی که شبیه شماره حساب اصلی هستند ولی برای ما فاقد ارزش باشند. نمونه بالا که شامل ۱۴ صفر است نشانگر شماره حساب واقعی برای MasterCard نیست ولی نشان‌دهنده الگوی آن است. به‌منظور جلوگیری از false positive، می‌توان از چندین روش تایید استفاده کرد که توسط نرم‌افزار اسکن کننده صورت می‌پذیرد. یکی از این روش‌ها، اعتبارسنجی Mod 10 است. آخرین رقم شماره حساب کارت اعتباری همیشه checksum است و می‌توان با Mod 10 محاسبه شده برای مابقی اعداد مقایسه کرد. عبارات منظم برای شماره حساب اصلی پرداخت‌های کلان در جدول ۴-۱۵ آمده است.

جدول ۴- ۱۵: عبارات منظم برای شماره حساب اصلی پرداخت‌های هنگفت

برند پرداخت	عبارت منظم
MasterCard	^5[1-5][0-9]{14}5
American Express	^3[47][0-9]{13)$
Diners Club	^3(?:0[0-5]|[68][0-9])[0-9]{11}$
Discover	^6(?:01115(0-9)(2))(0-9){12}$
JCB	^(?:2131|1800|35\d{3})\d{11}$
Visa	^4[09}{12}(?:[0-9]{3})?$

[۴۱] Regular Expressions

نقض امنیت[۴۲]

فعالیت‌های لیست شده در جدول ۴-۱۶ را می‌توان فقط در یک عبارت " نقض امنیت" خلاصه کرد. عبارت نقض امنیت به‌شکل گسترده در صنعت و امنیت مورد استفاده می‌گیرد و مجموعه‌ای از وقایع است که منجر به افشای اطلاعات مهم صاحب کارت می‌شود. شرکت‌ها معمولا مایل به افشای نقض‌های امنیتی خود نمی‌باشند چون نمی‌خواهند در معرض توجه همگانی قرار بگیرند.

جدول ۴-۱۶: مراحل نقض امنیتی داده‌های کارت

مرحله	اقدام	مدت
۱. جمع‌آوری اطلاعات	کسب اطلاعات در مورد کنترل‌های مربوط به فروشگاه‌های کوچک کسب اطلاعات در مورد تکنولوژی برنامه پرداخت و آسیب‌پذیری‌های هدف	چندین روز تا چندین هفته
۲. آماده‌سازی بدافزار	وابسته به اطلاعات مربوط به برنامه پرداخت، شخصی سازی بدافزار موجود یا ایجاد بدافزار جدید.	چندین روز تا چندین هفته
۳. نفوذ به محیط فروشگاه	شکستن کنترل‌های فیزیکی یا منطقی فروشگاه	چندین ساعت تا چندین روز
۴. جمع‌آوری اطلاعات مهم	جمع‌آوری اطلاعات از حافظه، دیسک ذخیره یا ارتباطات و ارسال آن به رایانه مهاجم	چندین ساعت تا چندین سال
۵. فروش	مهاجم‌ها با فروش dumpها کسب درآمد می‌کنند	چندین ساعت تا چندین هفته
۶. کسب درآمد	برای کارت‌های اعتباری: • انجام خریدهای برخط در فروشگاهای کوچک. • فروش کالاهای خریداری شده و بدست آوردن پول نقد برای کارت‌های بانکی (PIN):	چندین ساعت تا چندین سال

[۴۲] Security Breach

	• برداشت پول نقد از ATM	
۷. افشا	افشای عمومی (مشتریان درباره کلاهبرداری‌های تراکنش می‌دهند) آسیب‌های داخلی (رفتار غیرعادی سیستم) گزارش	چندین روز تا چندین ماه

فصل پنجم:

نفوذ به نواحی آزاد امنیتی

استانداردهای صنعت کارت پرداخت چالش‌هایی را برای کنترل‌های امنیتی فرآیند پرداخت به وجود آورده است. این استانداردها در حال تامین سیستم‌های امن هستند، ولی در داخل نرم‌افزار یا سخت‌افزار اجازه ایجاد آسیب‌پذیری‌ها را که توسط طراحی به وجود آمده است را می‌دهد و بدین ترتیب تاجران به امید یک منبع قابل اعتماد، ملزم به پیاده سازی کنترل‌های امنیتی می‌شوند، و این امر اجازه نقض امنیتی را به مهاجمان خواهد داد. نمونه‌ای از چنین عملکردی داده‌های محافظت نشده در حافظه، ترافیک شبکه رمزگذاری نشده است.

حافظه برنامه پرداخت

مهاجم‌ها در داخل شبکه تاجران، بدافزار تجزیه کننده حافظه که براساس سیستم‌های ویندوز یا سرورهای غیرعمومی است را نصب می‌کنند تا بتوانند همه داده‌های نوار مغناطیسی را بدست آورند.

تحلیل حافظه

تحلیل حافظه ، روشی برای جمع‌آوری اطلاعات حساس داده‌های کارت پرداخت است که توسط فرآیند برنامه پرداخت به‌وجود آمده است. تحلیل حافظه برای بدافزارهایی استفاده می‌شود که با ربودن اطلاعات کاربران از حافظه به برنامه‌های پرداخت حمله می‌کنند. اکثر برنامه‌های پرداخت بمنظور ارائه خدماتی برای مسیریابی، فرایند پرداخت و خدمات مشتریان، داده‌های زیادی را به صورت متن ساده بدون رمزگذاری قرار می‌دهند. بنابراین بایستی به یکی از این روش‌های موجود اعتماد کند تا بتوان داده‌ها را بصورت نقطه به نقطه رمزگذاری کند. بنابراین فروشندگان نرم‌افزار پرداخت هیچ استاندارد اجباری برای محافظت از حافظه برنامه پرداخت خود ندارند اما در عوض تاجران مجاب به محافظت از حافظه رایانه‌های کاربران خود از قبیل کنترل‌های فیزیکی و شبکه هستند.

WinHex

مشاهده حافظه ویندوز در هنگام استفاده از ابزار خاص امکان‌پذیر است. WinHex چنین ابزاری است که برای تست امنیتی و تحقیقات جرم‌یابی استفاده می‌شود. ارزیاب امنیتی معتبر در حین ارزیابی‌های PA-DSS, PCI DSS و با اجرای انواع مختلف تراکنش که همراه با شماره کارت از پیش تعریف شده است، کل دیسک را به‌منظور پیدا کردن این شماره‌های تعریف‌شده جستجو می‌کند و اگر در حین جستجو یک شماره حساب اصلی در یک متن ساده (بدون رمزگذاری) پیدا کند هیچ بازخوردی ارائه نمی‌دهند چون استانداردهای صنعت کارت پرداخت ملزم به حفاظت حافظه نیست.

استفاده از MemoryScraper

به‌منظور فیلتر کردن دنباله بایت‌های حافظه برای یافتن شماره حساب اصلی و مسیر داده کارت‌های پرداخت، از نرم‌افزار MemoryScraper به کمک عبارات منظم استفاده می‌کنند. زمانی‌که عبارت منظم به یک رشته از داده اعمال می‌شود، بخشی از حافظه که برای جستجو داده‌های حساس اختصاص داده می‌شود باید از فرمت باینری به رشته تبدیل شود. چندین روش مختلف رمزگذاری داده برای شماره حساب اصلی و هدایت داده در حافظه داخلی وجود دارد که توسط برنامه‌های پرداخت مورد استفاده قرار می‌گیرد و چندین محیط توسعه برای ذخیره یک رشته یکسان با فرمت‌های مختلف وجود دارد از جمله: ASCII, Unicode. برای مثال .NET. بطور پیش فرض از Unicode و یا Delphi, C++ از ASCII استفاده می‌کند. که در این‌صورت یک رشته ASCII هرکاراکتر را در یک بایت ذخیره می‌کند ولی در مقابل یک رشته Unicode هر کاراکتر را در دو بایت ذخیره می‌کند (۱-۵).

جدول ۵-۱: کدهای Unicode و ASCII برای ارقام و مسیر جداکننده‌های نواحی داده‌ها

کاراکتر	کد ASCII (decimal)	کد ASCII (hexadecimal)	Unicode (decimal)	Unicode (hexadecimal)
0	48	30	0 48	00 30
1	49	31	0 49	00 31
2	50	32	0 50	00 32
3	51	33	0 51	00 33
4	52	34	0 52	00 34
5	53	35	0 53	00 35
6	54	36	0 54	00 36
7	55	37	0 55	00 37
8	56	38	0 56	00 38
9	57	39	0 57	00 39
=	61	3D	0 61	00 3D
^	94	5E	0 94	00 5E

تنها تفاوت در این است که کاراکترهای Unicode شامل یک بایت صفر اضافی است. در جدول ۵-۲ این تفاوت برای یک شماره حساب اصلی نشان داده شده است. داده‌های واقعی در حافظه معمولا بصورت hexadecimal نمایش داده می‌شود و در کد مرجع، اعداد hexadecimal با پیشوند "0x" شروع می‌شود که

می‌توان با اعداد decimal که با پیشوند "0" شروع می‌شود را از هم تشخیص داد. برای نمونه کارکتر "0" در ASCII کد ۴۸ (decimal) یا کد 0x30 (hexadecimal) را دارد.

جدول ۵-۲: رمزگذاری ASCII و Unicode برای یک نمونه از شماره حساب اصلی

رمزگذاری	رشته	داده‌های واقعی در حافظه بصورت (hexadecimal)	طول (بایت)
ASCII	4005554444444403	34 35 35 35 30 30 34 34 34 34 34 34 34 34 33 30	16
Unicode	4 0 0 5 5 5 4 4 4 4 4 4 4 4 0 3	00 34 00 30 00 30 00 35 00 35 00 35 00 34 00 34 00 34 00 34 00 34 00 34 00 34 00 34 00 30 00 33	32

چگونگی عملکرد جستجوکننده عبارت منظم

عبارات منظم مجموعه‌ای از پارامترها یا الگوها را برای جستجو بکار می‌گیرد. در اینجا پارامترهای موردنظر شماره حساب اصلی، مسیر۱، و مسیر۲ است که به منظور ایجاد عبارت منظم نهایی به دستورالعمل‌های خاصی تبدیل شده‌اند (جدول ۵-۳).

جدول ۵-۳: دستورالعمل عبارت منظم که برای جستجو مسیرها و شماره حساب اصلی بکار می‌رود

دستورالعمل	توضیح	کاربرد
\s?	\s معرف کاراکتر فاصله خالی ? معرف عدم وجود مولفه یا فقط یک‌بار	4[0-9]{3}\s?[0-9]{4}\s?[0-9]{4}\s?[0-9]{4} تمام شماره حساب‌هایی که در گروه ۴رقمی هستند را پیدا می‌کند، برای مثال: 4005 5544 4444 4403
[x-y]	معرف محدوده ارقام مابین x و y	مثال ۱: 5**[1-5]**[0-9]{14} همه کارت‌هایی که با پیشوند ISO هستند را پیدا می‌کند، 51، 52، 53، 54، و 55 مثال ۲: [0-2][0-9][0-1][0-2] همه تاریخ انقضاها با فرمت سال، ماه را پیدا می‌کند، برای مثال 1402
{n}	معرف تعداد رقم‌های مولفه قبلی که به تعد n بار است.	5[1-5]**[0-9]{14}** پس از پیشوند، قسمت اصلی PAN را پیدا می‌کند، برای مثال: 54**99830000000601**

x	معرف خود رقم x .	5[1-5][0-9]{14} هر PAN را با پیشوند ۵ پیدا می‌کند، برای مثال: 5499830000000601
+	معرف مولفه با یک‌بار یا بیش از یک‌بار	\^.+\^ نام صاحب کارتی را از مسیر۱ پیدا می‌کند که طول متغیر داشته و مابین دو جداساز " ^" قرار گرفته است، برای مثال: 4005554444444403^**GOMZIN/SILVA**^1512

شماره حساب اصلی یک طول مشخصی دارد و فقط شامل ارقام بوده و با پیشوند مشخصی شروع می‌شود. این ویژگی‌ها به یک مجموعه از دستورات عبارت منظم مبدل شده است:

```
(?:4[0-9]{12}|4[0-9]{3}\s?[0-9]{4}\s?[0-9]{4}\s?[0-9]{4}|5[1-5][0-9]
{14}|6(?:011|5[0-9][0-9])[0-9]{12}|3[47][0-9]{13}|3(?:0[0-5]|[68][0-9])
[0-9]{11}|(?:2131|1800|35\d{3})\d{11})
```

علاوه بر الگوهای شناسایی شماره حساب اصلی، مسیرهای ۱و ۲ نیز ساختار ثابتی از مجموعه پارامترهای از پیش تعریف شده دارد: شماره حساب اصلی به‌اضافه تاریخ انقضای چهار بایتی که توسط کاراکتر "=" از هم مجزا شده است:

```
(?:(?:\=[0-2][0-9][0-1][0-2]|\^.+\^[0-2][0-9][0-1][0-2]))
```

مقابله با False Positives

یکی از مشکلاتی که به‌هنگام استفاده از عبارات منظم به‌منظور جستجوی شماره حساب اصلی می‌توان با آن مواجه شد، False Positives (مجموعه‌ای از اعداد تصادفی که کاملا شبیه به شماره حساب اصلی است) می‌باشد. بمنظور جلوگیری از چنین نتایج بدی چندین روش وجود دارد.

ابتدا باید به عبارت منظمی که برای جستجوی شماره حساب اصلی بکار می‌بریم دقت کافی داشته باشیم. ابتدا و انتهای آن (?<!) و (?!) است. برای مثال، عبارت منظم 4[0-9]{12} یک شماره حساب اصلی ۱۳رقمی پیدا خواهد که همیشه با عدد ۴ شروع و با ۱۲رقم بعد از آن ادامه می‌یابد. بدین ترتیب، برای جستجوی یک‌باره انواع مختلف کارت‌ها، چندین عبارت منظم می‌تواند با هم ترکیب شود:

```
string pan_pattern =

 @"(?<![\d])(?:4[0-9]{12}|4[0-9]{3}\s?[0-9]{4}\s?[0-9]{4}\s?[0-9]
{4}|5[1-5][0-9]{14}|6(?:011|5[0-9][0-9])[0-9]{12}|3[47][0-9]
{13}|3(?:0[0-5]|[68][0-9])[0-9]{11}|(?:2131|1800|35\d{3})\d{11})
(?![\d])";
```

عبارات (?<![\d]) و (?![\d]) بدین منظور اضافه شده است تا مشخص کند که شماره حساب اصلی قسمتی از مجموعه اعداد تصادفی یافت شده نبوده و ارقامی قبل و یا بعد از شماره حساب اصلی وجود ندارد.

عبارت test_pattern مورد استفاده در test_rgx، یکی دیگر از روش‌های مقابله با false positive است:

```
string test_pattern =
 @"(?:0{7}|1{7}|2{7}|3{7}|4{7}|5{7}|6{7}|7{7}|8{7}|9{7})";
```

شماره حساب اصلی کارت‌های تست شده که توسط برندهای پرداخت ایجاد شده و توسط فروشندگان نرم‌افزار پرداخت مورد استفاده قرار گرفته است، اغلب یک رقم یکسانی را معمولا ۷ بار یا بیشتر تکرار کرده‌اند (جدول ۵-۴).

جدول ۵-۴: نمونه‌هایی از کارت‌های تست False Positive

شماره حساب اصلی کارت تست شده	نوع کارت
۴۰۰۵۵۵۴۴۴۴۴۴۴۰۳	Visa
۴۴۸۵۵۳۰۰۰۰۰۰۰۱۲۷	Visa
۵۴۹۹۸۳۰۰۰۰۰۰۰۶۰۱	MasterCard
۵۵۶۷۳۰۰۰۰۰۰۰۰۰۱۶	MasterCard
۳۷۱۱۱۱۱۱۱۱۱۱۱۱۴	American Express

عبارت test_rgx هر شماره کارتی که شامل ۷ رقم تکرار یا بیشتر باشد را فیلتر خواهد کرد:

```
test_rgx = new Regex(test_pattern, RegexOptions.IgnoreCase);
MatchCollection test_matches = test_rgx.Matches(result.PAN);
if (test_matches.Count > 0)
    break;
```

بررسی رقم-بررسی Mod 10 نیز یک روش دیگر می‌باشد:

```
if                                                    (!PassesLuhnTest(result.PAN))
   break;
```

شنود

شنود داده بعد از تجزیه حافظه یکی دیگر از خطرناک‌ترین تهدیدات امنیتی برای برنامه‌های پرداخت است.

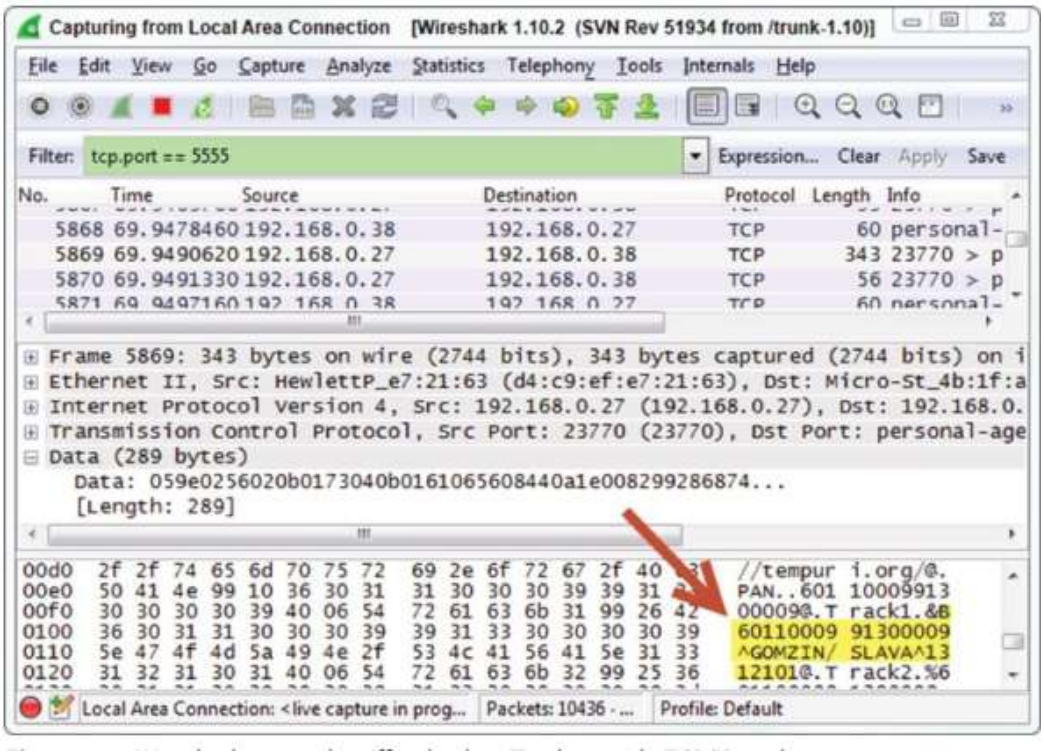
ترافیک شبکه‌های محلی

چندین روش برای به‌دست آوردن داده‌های در حال انتقال وجود دارد. یکی از روش‌های شنود، نصب وسیله شنود شبکه مخفی به شبکه فروشگاه است. وسیله شنود کننده همه ترافیک شبکه را شنود کرده و آن را به مرکز کنترل از راه دور ارسال می‌کند.

این نوع از حمله برای معماری دومنظوره POS/Store Server خیلی خطرناک است چون این سرورها اطلاعات پرداخت صاحب‌کارت را بمنظور پردازش بیشتر از پایانه فروش به سرور فروشگاه ارسال می‌کند.

شنودکننده‌های شبکه

نرم‌افزار خاصی تحت عنوان network sniffer یا packet analyzer برای شنود ترافیک شبکه استفاده می‌شود. اگر محموله‌ها رمزگذاری نشده باشد، مشابه روش گفته شده در بخش‌های قبلی برای تجزیه حافظه اطلاعات حساس، براحتی می‌تواند فیلتر شود. Wireshark یک نمونه از شنودکنندگان شبکه است. شکل ۵-۱ بسته‌های TCP/IP را مابین پایانه فروش (مشتری) و سرور نشان می‌دهد و مسیر۱ در یک متن ساده (بدون رمز) در داخل محموله TCP قرار دارد.



شکل ۵-۱: شنودکننده Wireshark مسیر ۱ را در داخل بسته TCP/IP نشان می‌دهد

بهره برداری از آسیب‌پذیری‌های دیگر

حملاتی علیه سخت‌افزار هم صورت می‌گیرد که محرمانگی داده‌های حساس صاحب‌کارت را هدف قرار نداده و درستی[۴۳] و دسترس‌پذیری[۴۴] برنامه پرداخت را هدف می‌گیرد.

حمله به برنامه پرداخت

چندین هدف برای حمله به برنامه پرداخت وجود دارد ولی می‌توان تنها در یک هدف بهره‌برداری از ضعف‌های کد برنامه آن‌را خلاصه کرد. تفاوت مابین حملات اصلی (مانند تجزیه حافظه و شنود شبکه) و حمله به کد برنامه در این است که برای انجام حمله به کد برنامه، مهاجم باید اطلاعات بسیار زیادی راجع به برنامه داشته باشد، ولی حملات اصلی برای هر برنامه پرداختی کاربرد دارد. در ادامه انواع حمله به برنامه‌های مختلف ارائه شده است:

- سوء استفاده از API برنامه پرداخت
- حمله به تنظیمات
- حمله به به‌روزرسانی‌های نرم‌افزار
- Disassemble کردن کد برنامه

[۴۳] Integrity
[۴۴] Availability

- جعل اختیارات (گواهی‌ها) مشتری

حمله به سخت‌افزار پرداخت

نفوذ به نرم‌افزار پایانه فروش خیلی قابل اعتماد است، اما تنها راه ربودن اطلاعات صاحب‌کارت نیست. سخت‌افزار بعنوان بخش مهمی از فرایند پرداخت است، و می‌توان براحتی به آن نفوذ کرد. در ادامه چندین حمله مختلف به سخت‌افزار ارائه شده است:

- تزریق کدهای آلوده به دستگاه‌های POI
- نصب دستگاه‌های جعلی به POI، بمنظور جعل عملکردهای نوار مغناطیس‌خوان و صحفه‌کلید.

هدف‌گیری تکنولوژی‌های جدید

همزمان با پیشرفت فرآیند پرداخت و روش‌های محافظت از آن، روش‌های نفوذ جدیدی هم پدیدار می‌شود. در حال حاضر اطلاعات کافی برای حملات خاص دردسترس نیست، چون این تکنولوژی‌های جدید هنوز به‌طور گسترده پیاده سازی نشده است و با استفاده از روش‌های سنتی از قبیل تجزیه حافظه و شنود می‌توان براحتی اطلاعات صاحب‌کارت را به‌سرقت برد. و بدین ترتیب به معرفی دو روش بالقوه نفوذ به سیستم‌های پایانه فروش در زمان آینده بسنده می‌کنیم:

- سخت‌افزار رمزگذاری نقطه به نقطه
- پرداخت‌های مربوط به دستگاه تلفن همراه
- چیپ‌های EMV که توسط نوارهای مغناطیسی مورد استفاده قرار می‌گیرد

احراز هویت صوتی جعلی از طریق تلفن‌همراه

این نوع حمله به درستی سیستم، از ضعف‌های موجود در استاندارد Force Post (و با نام احراز هویت از طریق تلفن‌همراه معروف است) بهره‌برداری می‌کند. منظور از Force Post این است که تراکنش با وجود لغو از طرف پردازشگر پرداخت باز هم می‌تواند توسط تاجر به زور انجام پذیرد. و این‌گونه احراز هویت معمولا توسط گوشی همراه صورت می‌گیرد (به همین علت با نام احراز هویت توسط گوشی‌همراه شناخته می‌شود). زمانی‌که کارت اعتباری بعد از کشیدن کارت در پایانه فروش منع می‌شود، حسابدار به منظور انجام دستی فرآیند با شماره موجود در پشت کارت اعتباری با بانک تماس می‌گیرد و از طرف بانک بعد از بررسی حساب صاحب‌کارت، کد احراز هویتی (معمولا ۶ رقم یا حرف) به حسابدار داده می‌شود تا بصورت دستی شماره کارت را وارد کند. در این‌صورت چندین ضعف امنیتی موجود خواهد داشت که معمولا کد احراز هویت از طریق گوشی همراه بصورت برخط اعتباری سنجی نمی‌شود، بنابراین می‌توان هر ترکیبی از ارقام و حروف را وارد کرد.

اجبار به احراز هویت آفلاین

هنگامی‌که شبکه یا میزبان فرآیند پرداخت از کار می‌افتد، سیستم‌های پایانه فروش احراز هویت را در حالت آفلاین ادامه می‌دهند و تراکنش‌های پرداخت بصورت محلی و توسط برنامه پرداخت و بدون اعتبارسنجی واقعی اطلاعات صاحب کارت صورت می‌گیرد. بنابراین موقع کشیدن کارت در پایانه فروش، از مرحله اصلی اعتبار سنجی صرف‌نظر می‌شود. و اگر بتوان اتصال شبکه را قطع کرد می‌توان از این ویژگی نهایت استفاده را کرد. برای مثال، اگر یک فروشگاه کنترل از راه دور از آنتن ماهواره‌ای برای ارتباط با شبکه پرداخت استفاده کند، می‌توان با پوشیدن نوار فویل به دور گیرنده، شبکه را مختل کرده و آن را به حالت ذخیره و ادامه مبدل کرد.

فصل ششم:

نفوذ به نواحی محفاظت شده با PCI

امروزه بیشتر فروشندگان برنامه‌های پرداخت، نرم‌افزار تایید شده PA-DSS را عرضه می‌کنند، و بیشتر تاجران، با PCI DSS سازگار هستند. اگر فرض کنیم که نواحی آسیب‌پذیری که توسط قوانین PCI محافظت شده کمتر از بقیه نواحی محافظت نشده آسیب‌پذیر هست، معقول خواهد بود. ولی با این وجود حفره‌های امنیتی نیز در داخل نواحی محافظت شده وجود دارد.

نواحی مورد توجه PCI

بخاطر دلایل تجربه شده، استانداردهای PCI در جاهایی حضور دارد که احتمال از دست رفتن داده‌های حساس وجود دارد. در فصل ۳ دو ناحیه کلیدی یعنی: داده‌های ذخیره‌شده (غیرفعال) و داده‌های در حال انتقال مورد بحث قرار گرفت. البته امنیت داده‌های ذخیره‌شده از نظر PCI کامل است ولی برای داده‌های در حال انتقال خیلی محدود است که در جدول ۶-۱ خلاصه شده است.

جدول ۶-۱: محافظت برنامه پرداخت توسط PCI

ناحیه آسیب‌پذیری	زیر-ناحیه	محافظت اجباری	نیازمندی PCI DSS
داده در حافظه		-	
داده ثابت	ذخیره موقت	•	نیازمندی ۳: محافظت از داده صاحب‌کارت
	ذخیر طولانی مدت	•	نیازمندی ۳: محافظت از داده صاحب کارت
	فایل‌های گزارش	•	نیازمندی ۳: محافظت از داده صاحب کارت
داده در حال انتقال	ارتباطات محلی	-	
	ارتباطات مابین دستگاه POI و POS	-	
	ارتباط با پردازشگر	○	نیازمندی ۴: رمزگذاری داده در حال انتقال در شبکه‌های عمومی و باز
کد و تنظیمات برنامه	کد برنامه	-	
	تنظیمات برنامه	-	

داده‌های ذخیره‌شده (غیرفعال)

داده‌های ذخیره‌شده یک اصطلاح امنیتی برای اطلاعات ذخیره شده روی دیسک سخت است و برای یک متن خاص، در تضاد با داده در حال انتقال مابین دستگاه و رایانه است. قبل از PCI، اطلاعات حساس در دیسک‌های سخت و رمزگذاری نشده ذخیره می‌شد، و فقط توسط شبکه تاجران و کنترل‌های فیزیکی محافظت می‌شد، که اغلب ضعیف یا اطلاعات کلا ناپدید می‌شدند. زمانی‌که PCI معرفی شد، فروشندگان نرم‌افزار پرداخت با انجام رمزگذاری، شروع به محافظت از اطلاعات ذخیره شده خود کردند که بهره‌برداری آسیب‌پذیری‌ها را به‌شدت کاهش داد.

ذخیره‌سازی موقت

بعضی از فروشندگان برنامه پرداخت بر این عقیده استوار هستند که هیچ داده حساسی را در پایانه فروش ذخیره نمی‌کنند، و در نتیجه سیستم‌های آن‌ها در مقابل حمله به داده‌های ثابت مصون هستند. چنین باورهایی، می‌تواند اشتباه باشد. تولید کنندگان، به ذخیره سازی، بعنوان نگه‌داری دراز مدت نگاه می‌کنند، مانند آرشیو پایگاه داده، درحالی‌که توسعه دهندگان اطلاع دارند که به‌هنگام ذخیره موقت اطلاعات چندین شرایط مختلف ممکن است به وجود بیاید. بهترین مثال برای چنین ذخیره‌سازی موقت S&F و TOR است که می‌تواند بعنوان جداول پایگاه داده محلی و فایل‌های ساده داده پیاده سازی شود. ویژگی مهم این دو مثال در لزوم انجام فرآیند رمزگذاری و رمزگشایی در همان دستگاه است چون به هنگام ذخیره، اطلاعات در دیسک سخت رمزگذاری می‌شود و به‌هنگام بازیابی باید دوباره رمزگشایی شود.

گزارش‌های برنامه

ثبت فایل‌های برنامه پرداخت اطلاعات بسیار جالبی برای مهاجمان خواهد داشت. برخلاف عملکردهای اصلی برنامه که توسط تولیدکنندگان ارائه شده است، ثبت فایل‌ها و محتوای آن توسط توسعه دهندگانی که اغلب راجع به امنیت اهمیتی قائل نیستند، مدیریت می‌شود. توسعه‌دهندگان می‌خواهند تا جایی‌که امکان‌پذیراست در مورد زمان اجرای برنامه مورد نظر اطلاعات کسب کنند تا بتوانند مشکلات را از راه دور حل کنند و فایل‌ها را با جزئیات بسیار زیادی ثبت و گزارش کنند. که این جزئیات ممکن است شامل مسیرهای کارت و شماره حساب باشد.

یکی دیگر از تهدیدات مرتبط با گزارش، استفاده از تابع درهم‌ساز شماره حساب اصلی و تولید نشانه است. در نگاه اول ممکن است که نشانه‌ها امن بنظر برسند اما توصیه می‌شود که شماره حساب واقعی جایگزینی شود.

یافتن فایل‌های گزارش و ذخیره‌سازی موقت

ذخیره‌سازی و گزارش موقت معمولا در یکی از شکل‌های زیر انجام می‌شود:

- پایگاه داده
- فایل‌های ساده
- فایل‌های متنی

پایگاه داده معمولا با گذرواژه محافظت می‌شود. اگرچه، اغلب برنامه‌های پرداخت معروف از مقادیر پیش فرض یا گذرواژه ذخیره شده در متن بدون رمز استفاده می‌کنند (جدول ۶-۲).

جدول ۶-۲: نمونه‌ای از حساب‌ها و گذرواژه‌های پیش فرض برای پایگاه داده

پایگاه داده	نام ورود	گذرواژه
Interbase	SYDBA	Masterkey
Microsoft SQL Server	Sa	
Oracle	System/manager	Sys/change_on_install
MySQL	Root	

شماره حساب اصلی درهم‌سازی شده

کاربرد استاندارد توابع درهم‌ساز بمنظور ارائه شکل کوتاهی از پیام یا فایل است. تابع درهم‌ساز با نام رمزنگاری یک‌طرفه هم شناخته می‌شود، چون از لحاظ ریاضی بازسازی دوباره پیام درهم‌سازی شده امکان‌پذیر نیست. از این ویژگی درهم‌ساز برای رمزنگاری شماره حساب کارت اعتباری استفاده می‌شود که دیگر نیازی به کلید رمزگشایی نیست. جدول ۶-۳ تفاوت مابین توابع مختلف هش را برای رمزگذاری شماره حساب اصلی نشان می‌دهد.

جدول ۶-۳: نمونه‌هایی از توابع هش مربوط به شماره حساب اصلی

تابع هش	اندازه	مثال
PAN در متن ساده	۱۶	۴۰۰۵۵۵۴۴۴۴۴۴۴۰۳
MD5	32	73bd8d04cc59610c368e4af76e62b3f1
SHA-1	40	f9b5eededb928241974368cbd97c055141813970
SHA-256	64	f85d4630aabe6d0d037ccb0ec0d95429aef6927b1e45a26da62cbd0bc344f6b4

جداول Rainbow و Brute Forcing

مشکل درهم‌سازی کردن شماره کارت اعتباری در این است که این اعداد قابل پیش‌بینی هستند. علاوه بر این، ۶ یا ۴ رقم اول این درهم‌ساز کد اغلب در یک متن بدون رمز قابل رویت است، که توسط استانداردهای

PCI یا دیگر استانداردها این اجازه داده شده است، و در واقع ما باید ۶ رقم را (۴-۶-۱۶) حدس بزنیم (جدول ۶-۴). از این فاکتورها می‌توان در حملات brute-force استفاده کرد که در این نوع از حمله از تمام ترکیب‌های ارقام شماره حساب استفاده می‌شود.

جدول ۶-۴: نمونه‌هایی از مخفی کردن شماره حساب اصلی با استفاده از قوانین PCI

نوع کارت	PAN ماسک شده	تعداد ارقامی که باید حدس زد
Visa	400555******4403	6
MasterCard	557552******7645	6
Amex	379640*****1007	5

جداول Rainbow همان نتایج درهم‌ساز از پیش تعیین شده است که به‌صورت دینامیک با درهم‌ساز کد مورد نظر مقایسه می‌شود. جدول ۶-۵ قسمتی از جدول محاسبه شده برای SHA-1 را نشان می‌دهد که برای شماره حساب کارت اعتباری اصلی محاسبه شده است.

جدول ۶-۵: نمونه‌هایی برای قسمتی از جدول Rainbow

شماره حساب اصلی	SHA-1
…	…
4005554444444395	e690e41f949423016e9346df2b4e7d6eb205c3b6
4005554444444403	f9b5eededb928241974368cbd97c055141813970
4005554444444411	96e8bddce2202451c828d57be33c94ae747f3594
…	…

ذخیره غیر امن کلیدهای رمزنگاری

مهم‌ترین قسمت استانداردهای PCI مربوط به حفاظت از اطلاعات ثابتی است که توسط فروشندگان نرم‌افزار و تاجران پیاده سازی می‌شود.

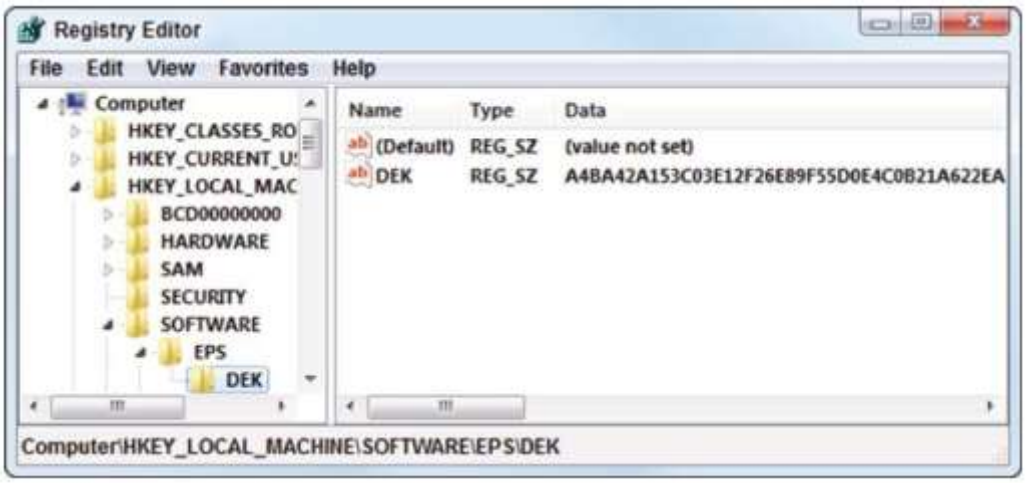
اگر POS قادر به رمزنگاری داده باشد، پس همیشه این احتمال هم وجود دارد که در همان دستگاه رمزگشایی هم صورت بگیرد. حتی اگر دستگاه برای این‌کار برنامه‌ریزی نشده باشد، چون کلید رمزگشایی در یک قسمتی از رایانه جای گرفته است.

گزینه‌های خیلی کمی برای مخفی‌کردن کلید موجود است که می‌توان در نرم‌افزار یا سخت‌افزار پنهان کرد. قسمت سخت‌افزاری خیلی گران‌قیمت است پس به ندرت می‌توان از این قسمت استفاده کرد ولی برای ذخیره کلید در حالت نرم‌افزاری گزینه‌های زیر را پیش‌رو داریم:

- درایو سخت (فایل داده)
- رجیستری
- کدهای غیرقابل تغییر (در فایل باینری برنامه)
- هر ترکیبی از روش‌های مذکور

کلید رمزنگاری داده و کلید رمزنگاری کلید

در یک رمزنگاری ساده، داده رمزنگاری شده با یک کلید ذخیره شده (با یکی از روش‌های بالا) رمزگشایی می‌شود. در بیشتر برنامه‌های پیچیده از روش KEK و DEK استفاده می‌شود که DEK در جایی در رایانه پنهان شده است و توسط KEK رمزگشایی می‌شود (شکل ۶-۱). مشکل این رهیافت در این است که KEK نیز باید در جایی در رایانه پنهان شود. پس وظیفه مهاجم همان پیدا کردن کلید مورد نظر است.



شکل ۶-۱: نمونه‌ای از AES 256 DEK رمزگذاری شده که در ریجستری ویندوز پنهان شده

چرخش کلید

چرخش کلید یک مکانیزمی است که برای حل دو مشکل زیر به‌وجود آمده است:

۱. جلوگیری از رمزگشایی کلید. هنگامی که مهاجم به تعداد زیادی از خروجی داده رمزگشایی شده (متن رمز) دست پیدا می‌کند، معمولا می‌تواند کلید را حدس بزند ولی با تغییر کلید رمزگشایی این امکان حدس وجود نخواهد داشت.
۲. به حداقل رساندن داده‌های فاش شده به هنگام کشف کلید رمزگشایی.

مشکل چرخش کلید در این است که پیاده سازی این ویژگی بسیار سخت و پیچیده است. و برای همین بسیاری از تاجران یک دستورالعملی تحت عنوان تغییر دستی کلید ارائه می‌دهند.

کلیدهای پیش فرض

مشکل این نوع کلیدها تقریبا بسته به مشکل چرخش کلید است. اگر فروشندگان برنامه در تولید کلید خودکار کم‌کاری کنند منجر به ایجاد کلیدهای پیش فرض یکسانی برای همه دستگاه‌های پایانه فروش خواهد شد.

دستیابی به کلیدها

به‌منظور دستیبابی به کلیدها، مهاجم باید بداند در کجا و چگونه این کلیدها ذخیره می‌شود که این اطلاعات می‌تواند توسط مهندسی معکوس یا افراد خودی انجام بگیرد.
افراد خودی کسانی هستند که قبلا یا در حال حاضر برای فروشندگان، تاجران نرم‌افزار پرداخت کار می‌کنند که اطلاعات و مستندات مهمی از طراحی و نحوه عملکرد کدها را دارند و می‌توان از اطلاعات آنان در دستیابی به کلیدها استفاده کرد.
مهندسی معکوس یا disassembling یک فرآیندی برای کدگشایی کد است. اکثر زبان‌های برنامه نویسی برای نوشتن برنامه پرداخت از یک مفسر استفاده می‌کنند که این مفسر اگر کدی را با زبان سطح پایین مثلا C, C++, Delphi تولید کند معکوس کردن این کد نوشته شده به کد مرجع مربوطه خیلی سخت خواهد بود. اما اگر با زبان‌های سطح متوسطی از قبیل Java, C# نوشته شده باشد می‌توان به‌راحتی کدمرجع را بدست آورد مگر اینکه کد مورد نظر به‌نحوی مبهم‌سازی شده باشد.

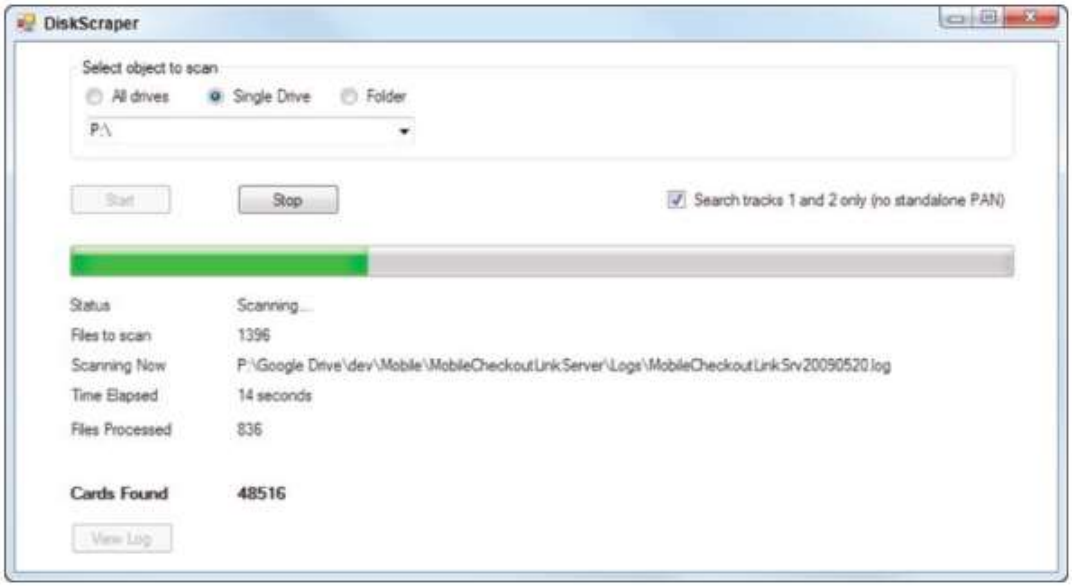
استفاده از DiskScraper

با این‌که رمزگذاری و مسیریابی داده‌های ذخیره‌شده یکی از مهترین مستلزمات صنعت کارت پرداخت است اما ممکن است افشای اطلاعاتی نیز رخ بدهد چون احتمال دارد کارمندی حذف آن اطلاعات را فراموش کند یا بطور ساده یک ضعف در نرم‌افزار پایانه فروش موجود باشد. چندین ابزار مختلف برای اسکن درایوهای سخت باهدف جستجوی داده‌های حساس وجود دارد که اگر این کار را انجام بدهیم در کمال تعجب مشاهده خواهیم کرد که شماره حساب‌ها و داده‌های مسیر که پاک شده بودند هنوز هم در سیستم موجود است. DiskScraper یک نمونه ساده است که وظیفه جستجوی فایل‌های شامل داده‌های حساس (.txt, .log, .dat) را برعهده دارد (شکل ۶-۲). این نرم‌افزار از روش جستجوی عبارات منظم استفاده می‌کند. با اجرای این برنامه همراه با نرم‌افزار پایانه فروش و شروع اسکن می‌توان گزارش نتایج را در یک فایل متنی مشاهده کرد (شکل ۶-۳).

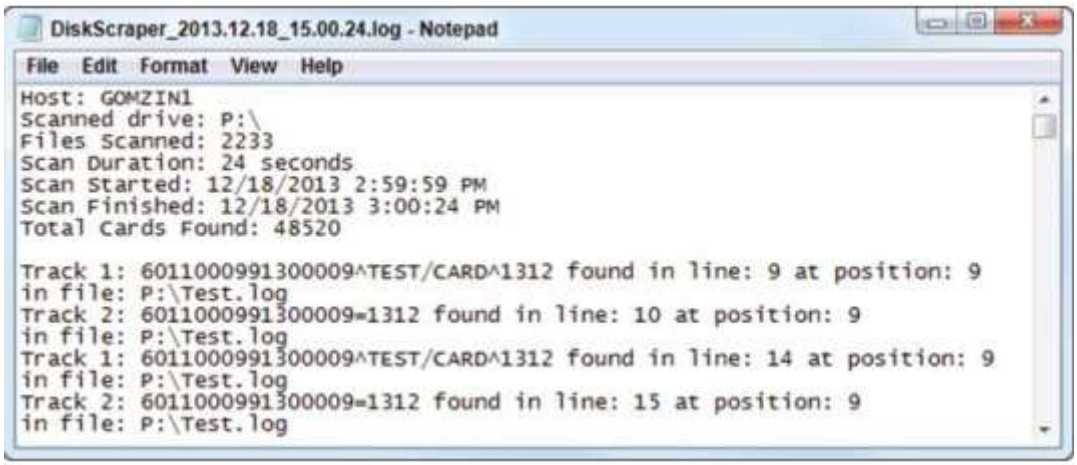
جستجوی بازگشتی[۴۵]

[۴۵] Recursive

اگر DiskScraper برای جستجوی کل سیستم تنظیم شود، در اینصورت این برنامه از روش بازگشت استفاده خواهد کرد که یک روش ساده و ظریف برای اسکن فایل می‌باشد. بدین صورت که تابع برگشت از پوشه ریشه شروع بکار کرده و خودش را تا زمان دستیابی به انتهای پوشه فراخوانی می‌کند (مسیر درخت را طی می‌کند).



شکل ۶-۲: استفاده از DiskScraper



شکل ۶-۳: نتایج جستجوی DiskScraper

فایل‌های متنی در مقایسه با فایل‌های باینری

دو نوع کلی فایل متن و باینری و با ساختار متفاوت داریم. فایل‌های متنی مجموعه‌ای از کاراکترهای ASCII یا Unicode است. خطوط متن توسط برگشت داده (0x0D) و خط جدید (0x0A) از هم جدا شده‌اند که جزئی از کاراکترهای ASCII هستند. فایل‌های باینری آرایه‌هایی از بایت‌ها هستند که ممکن است هر مقداری داشته باشند. با توجه به نوع فایل، چندین روش برای بازکردن و تکرار محتوای داخل آن وجود.

داده در حال انتقال: چه‌چیزی توسط PCI پوشش داده می‌شود؟

در مستلزمات محافظت از ارتباطات، استانداردهای PCI برحسب نوع شبکه: "باز، عمومی" و شبکه‌های دیگر با هم متفاوت هستند.

مشکل از آنجایی شروع می‌شود که شبکه‌های غیر "باز، عمومی" که شامل LAN داخلی، WAN مشترک، و frame مربوط به تاجر و پردازشگر است، نقش اساسی را در عملکرد برنامه‌های پرداخت ایفا می‌کنند. علاوه بر این برنامه‌های پرداخت از شبکه‌های مذکور برای انتقال داده‌های حساس هم استفاده می‌کند (جدول ۶-۶).

جدول ۶-۶: ملزومات محافظت از شبکه

نوع شبکه	مستلزم رمزگذاری با PCI می‌باشد؟	معمولا برای انتقال داده‌های حساس استفاده می‌شود
Internet	بله	مابین پایانه فروش و درگاه/پردازشگر پرداخت
Wireless	بله	مابین پایانه فروش و فروشگاه؛ سرور مابین پایانه فروش و درگاه/پردازشگر پرداخت
LAN	خیر	مابین پایانه فروش و سرور فروشگاه؛ مابین پایانه فروش و درگاه/پردازشگر پرداخت؛ مابین ماژول‌های برنامه پرداخت
WAN	خیر	مابین پایانه فروش و درگاه/پردازشگر پرداخت؛ مابین سرور فروشگاه و شرکت اصلی

آسیب‌پذیری‌های SSL

SSL یک پروتکل بسیار محبوب برای محافظت از داده‌های در حال انتقال می‌باشد. با این حال، SSL نیز غیرقابل شکست نبوده و ضعف‌های خود را دارد، مخصوصا زمانی‌که به‌طور مطلوب تنظیم نشده باشد.

نسخه‌های منسوخ شده SSL/TLS

SSL از زمانی‌که توسط Netscape به‌وجود آمده، چندین اصلاحیه به آن اضافه شده است، آخرین نسخه این پروتکل ۳,۰ بوده که از آن زمان به بعد همراه با نام TLS به کار خود ادامه داده است. و اگر برنامه پرداخت به آخرین بروزرسانی‌ها مجهز نشود ممکن است به حملات مختلفی آسیب‌پذیر باشد چون هر نسخه جدید آن شامل اصلاحیه‌های جدیدی خواهد بود.

سازگاری ضعیف با متن‌رمز

یکی از ویژگی‌های SSL، توانانی سازگاری انواع سطوح مختلف رمزگذاری مابین کلاینت و سرور است. و از طرفی، این ویژگی مقرون به‌صرفه است چون اجازه ارتباط مابین کلاینت و سرور را در هر سطح مختلفی می‌دهد. و این منجر به این می‌شود که مهاجم سطح رمزگذاری خود را کاهش داده و به ارتباطات ایمن مفروض شده، حمله کند.

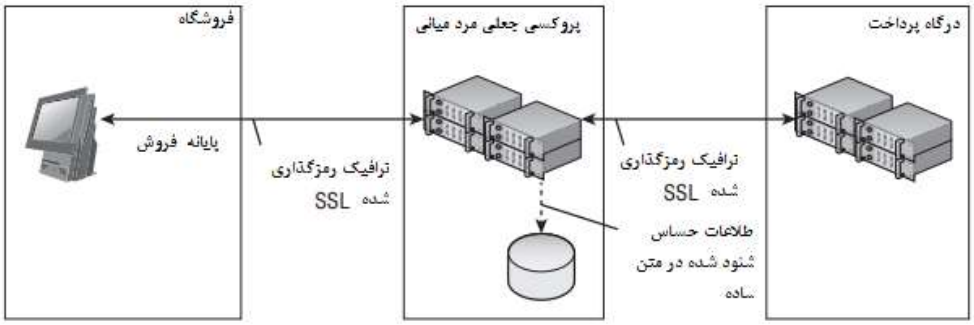
مرد میانی

مردمیانی[۴۶] یکی از معروف‌ترین نوع حملات است. و به مهاجم اجازه حمله به ترافیک شبکه مابین کلاینت و سرور را می‌دهد و بدین ترتیب مهاجم اطلاعات حساس کارت پرداخت را به‌سرقت می‌برد. تنظیمات کلاینت می‌تواند دست‌کاری شده و ارتباطی را مابین سرور جعلی و آدرس میزبان ایجاد کند و سرور نیز آدرس مورد نظر را مبنی بر معتبر بودن استفاده کند. اگر کلاینت آدرس میزبان و صادر کننده گواهی سرور را تایید نکند، فرض می‌شود که با یک سرور قانونی در حال صحبت است.

یکی دیگر از سناریوهای حمله مرد میانی، با وارد کردن پروکسی جعلی و با استفاده از گواهی سرور جعلی منجر به اختلال ترافیک شبکه مابین کلاینت قانونی و سرور معتبر می‌شود که در شکل ۶-۴ نشان داده شده است. سرور جعلی شروع به ایجاد ارتباط با سرور قانونی کرده و تظاهر به کلاینت قانونی می‌کند. این گفته تحت چندین شرایط ممکن است اتفاق بیافتد:

۱. مهاجم به صدور گواهی عمومی یا اختصاصی که از سرور اصلی صادر شده است دسترسی پیدا کند، و یک گواهی جعلی را از طرف صادرکننده اصلی صادر کند. برخی از تاجران و پردازشگرها ممکن است از گواهی‌های دست‌ساز که توسط گواهی‌های خاصی صادر شده استفاده کنند که کنترل‌های امنیتی ناکافی را دارد.
۲. مهاجم گواهی سرور را در مخازن گواهی کلاینت به‌عنوان گواهی قانونی مستقر می‌کند. کلاینت تصور می‌کند که گواهی سرور مورد نظر معتبر بوده و شروع به ارتباط با پروکسی جعلی می‌کند.

[۴۶] Man In the Middle



شکل ۶-۴: حمله مردمیانی

فصل هفتم:

مکانیزم‌های دفاعی

رمزنگاری در برنامه‌های پرداخت

زمانی که موضوع حفاظت از اطلاعات مطرح باشد، نیاز به استفاده مناسب از کاربردهای مختلف رمزنگاری مورد توجه قرار می‌گیرد. در مورد برنامه‌های پایانه فروش، داده‌های حساس دارندگان کارت باید از چشم دیگران در طول انجام فرآیند پرداخت به دور باشد. برنامه‌های پرداخت مدرن در حال حاضر از رمزنگاری در موارد بسیاری استفاده می‌کنند، با این حال همیشه در امن‌ترین روش مورد استفاده قرار نمی‌گیرند. بیشتر توسعه‌دهندگان نیز با اصل استفاده از پیاده‌سازی‌های مطرح الگوریتم‌های رمزنگاری به جای استفاده از پیاده‌سازی‌های تایید نشده خود آشنا هستند. مشکل این است که رمزنگاری تنها محدود به استفاده از یک کتابخانه پیاده‌سازی الگوریتم نیست و این تنها بخشی از مشکلات است. موضوع مدیریت کلید نیز وجود دارد، که در انواع مختلف رمزگذاری‌ها دیده می‌شود و در طراحی برنامه پرداخت نیازمند توجه خاصی می‌باشد.

در حالت کلی ۳ گروه توابع رمزنگاری متقارن[۴۷]، نامتقارن[۴۸] و رمزنگاری یک طرفه[۴۹] وجود دارد. هر گروه بر اساس مفاهیم ریاضی متفاوت طراحی شده و برای استفاده در کارهای مختلف در نظر گرفته شده‌اند. وجه تمایز مهم دیگری که در هنگام انتخاب نوع رمزگذاری باید مورد توجه قرار گیرد در مورد رابطه بین کلیدهای رمزگذاری و رمزگشایی می‌باشد. به شکل خلاصه، کلید متقارن برای هر دو مرحله رمزگذاری و رمزگشایی یکسان می‌باشد؛ الگوریتم‌های نامتقارن از کلیدهای متفاوتی برای رمزگذاری و رمزگشایی استفاده می‌کنند و تابع درهم‌ساز یک طرفه فاقد کلید است (البته چندین مورد استثنا وجود دارد). تنوع در نحوه مدیریت کلید زمینه‌های کاربرد عملی الگوریتم‌های مختلف را تعیین می‌کند.

همانگونه که در جدول ۷-۱ مشخص است، رمزگذاری متقارن مفیدترین الگوریتم رمزنگاری برای رمزگذاری داده‌ها در برنامه‌های پرداخت می‌باشد. با این وجود، انواع رمزگذاری معمولا به نوعی در نرم افزار POS دخیل هستند.

جدول ۷-۱: خلاصه مقایسه رمزگذاری متقارن، نامتقارن و درهم‌سازی یک طرفه

	متقارن	نامتقارن	درهم‌ساز یک طرفه
کلیدها	کلید یکسان برای رمزنگاری و رمزگشایی	دو کلید متفاوت: یکی برای رمزنگاری و دیگری برای رمزگشایی	فاقد کلید
رمزگشایی	متن رمز شده می‌توانند با	کلید متفاوتی برای	رمزگشایی و برگرداندن داده

[۴۷] Symmetric
[۴۸] Asymmetric
[۴۹] One-way Encryption

	استفاده از همان کلید رمزنگاری رمزگشایی شود	رمزگشایی متن رمز شده لازم است	رمز شده با درهم‌ساز یک طرفه امکان پذیر نیست
کارایی	بالا	پایین	بالا
موارد استفاده در رمزگذاری داده‌ها	❖ ذخیره موقت داده (S&F) ❖ سوابق تراکنش‌های طولانی مدت ❖ رمزنگاری داده‌های ارتباطی ❖ رمزگذاری نقطه به نقطه نرم‌افزاری	❖ رمزنگاری نقطه به نقطه نرم‌افزاری	❖ رمزگذاری رمز عبور
زمینه استفاده به عنوان بخشی از سامانه رمزگذاری	❖ رمزنگاری نقطه به نقطه سخت‌افزاری ❖ SSL	❖ امضای داده دیجیتالی ❖ امضای کد ❖ احراز هویت کاربر ❖ SSL	❖ امضای داده دیجیتالی ❖ امضای کد ❖ SSL
الگوریتم‌های نمونه	AES, TDES	RSA	SHA
اندازه کلید/digest قابل قبول (بیت)	AES: 128 TDES: 112	RSA: 1024	SHA: 160
اندازه پیشنهادی کلید/digest (بیت)	AES: 256 TDES: 168	RSA: 2048	SHA: 256

یکی از کاربردهای قابل توجه رمزنگاری امضای دیجیتالی می‌باشد. امضای دیجیتالی امکان احراز هویت نویسنده انواع داده‌های دیجیتالی مانند اسناد، پیام‌ها و کدهای نرم‌افزار را در اختیار ما قرار می‌دهد که یک ویژگی خیلی مهم در کار کردن با برنامه‌های پرداخت و تراکنش‌های مالی به حساب می‌آید. رمزنگاری نامتقارن و یک طرفه برای تولید امضا استفاده می‌شود. نکته مهمی که در امضای داده و به خصوص داده‌های حساس مالکان کارت باید بدانیم این است که امضای دیجیتال از محرمانگی محافظت نمی‌کند. بنابراین، علاوه بر امضا کردن، اقدامات امنیتی (رمزگذاری) دیگر جهت جلوگیری از افشای اطلاعات لازم است. با این حال، می‌توان از امضای دیجیتال برای محافظت از اطلاعاتی که از جهت افشا حساس نمی‌باشند برای جلوگیری از دستکاری توسط افراد غیرمجاز و بدافزارها استفاده کرد. برای مثال، فایل‌های تراکنش پایانه فروش (اطلاعات درباره کل تراکنش‌های مشتری که بین پایانه فروش، سرور ذخیره‌سازی و مرکز سازمان در چرخش است) می‌تواند گزینه مناسبی برای استفاده از امضای دیجیتال باشد.

استانداردهای رمزنگاری

همانگونه که صنعت پرداخت بیشتر و بیشتر بر روی امنیت هزینه می‌کنند، رمزنگاری سطوح پیشرفته نیز در فرآیند پرداخت کارتی مورد استفاده قرار می‌گیرد. امروزه، برنامه‌های پرداخت از کتابخانه رمزنگاری استفاده می‌کنند که چندین سال پیش تنها توسط متخصصان رمزنگاری مورد استفاده قرار می‌گرفت. راهکارهای رمزگذاری نقطه به نقطه از سخت‌افزار رمزنگاری بسیار امن برای رمزگذاری داده‌های حساس در پایانه مشتری و رمزگشایی در مرکز داده پردازش کننده پرداخت استفاده می‌کنند. در سالهای اخیر استانداردهایی توسط فعالان صنایع پولی و بانکی، توسعه دهندگان رمزنگاری و برنامه‌های امنیتی جهت تنظیم قوانین امنیت کارت پرداختی به وجود آمده‌اند. برای مثال در چندین سال پیش، FIPS 140-2 و TR-39 برای متخصصان امنیت دستگاه‌های ATM و تراکنش‌های PIN شناخته شده بودند.

مؤسسه ملی استاندارد و فناوری (NIST) به عنوان نقش محوری در توسعه استانداردهای رمزنگاری و همچنین صحت سنجی و اعتبارسنجی پیاده‌سازی‌های الگوریتم‌های رمزنگاری ایفا می‌کند. NIST چندین استاندارد پردازش اطلاعات فدرال (FIPS) توسعه داده است که نیازمندی‌های لازم برای سیستم‌ها و الگوریتم‌های رمزنگاری را تعریف می‌کند. یکی از محصولات معروف NIST استاندارد FIPS 197 است که در جهان با عنوان AES شناخته می‌شود. محصول دیگر FIPS 180-4 است که به عنوان SHA آن را میشناسیم. هر دو آنها به صورت گسترده در امنیت برنامه‌ها (که برنامه‌های پرداخت نیز شامل می‌شوند) و محافظت از داده‌ها استفاده می‌شوند.

جدول ۷-۲ الگوریتم‌های تایید شده توسط NIST را نشان می‌دهد. توجه نمایید که پذیرش الگوریتم به طول کلید (digest در صورت استفاده از درهم‌ساز) آن بستگی دارد. به عنوان یک قاعده کلی هرچه تعداد بیت بیشتری در کلید باشد، الگوریتم قوی‌تر می‌باشد. با این حال، AES با کلید 128 بیتی امن‌تر از Triple Des با کلید 168 بیتی است.

جدول ۷-۲: الگوریتم‌های رمزنگاری تایید شده توسط NIST

الگوریتم	استاندارد	طول کلید/digest (بیت)	استفاده
DES	FIPS 46-3	56	غیر مجاز
Single-length Triple-DES	-	56	غیر مجاز
Double-length Triple-DES	NIST SP 800-67	112	بعد از سال ۲۰۱۵ غیر مجاز
Triple-length Triple-DES	NIST SP 800-67	168	بعد از سال ۲۰۳۰ غیر مجاز
AES 128	FIPS 197	128	مجاز

AES 192	FIPS 197	192	مجاز
AES 256	FIPS 197	256	مجاز
SHA-1	FIPS 180-4	160	مجاز
SHA-224	FIPS 180-4	224	مجاز
SHA-256	FIPS 180-4	256	مجاز
SHA-384	FIPS 180-4	384	مجاز
SHA-512	FIPS 180-4	512	مجاز

سخت‌افزار رمزنگاری

تمام توابع رمزنگاری می‌توانند به صورت نرم‌افزاری پیاده‌سازی شوند. اما در عمل، بیشتر توابع رمزنگاری به واحدها و مولفه‌های سخت‌افزاری اختصاصی سپرده می‌شوند. دو وظیفه بسیار مهم برای سخت‌افزار رمزنگاری برنامه‌های پرداخت وجود دارد:

۱. انجام عملیات‌های رمزنگاری سنگین واحد پردازش مرکزی (رمزگذاری، رمزگشایی و مدیریت کلید) و با این کار بار پردازشی از کامپیوتر میزبان که برنامه‌های سمت کاربر و سمت سرور POS را اجرا می‌کنند، حذف می‌شود.
۲. ایجاد مانع بین شبکه‌ای که در دسترس هکرها بوده و شامل کامپیوترهای ناامن می‌شود و منطقه ایمن داخل سخت‌افزار که از طریق کنترل‌های فیزیکی و منطقی از محیط‌های خطرناک جدا شده است.

TRSM و HSM نمونه‌هایی از سخت‌افزار رمزنگاری است که اغلب در راهکارهای برنامه‌های پرداخت مورد استفاده قرار می‌گیرد:

- TRSM (Tamper-Resistant Security Module): یک ماژول سخت‌افزاری می‌باشد که در پایانه-های پرداخت جهت ذخیره و تولید کلیدهای رمزگذاری و انجام عملیات رمزگذاری استفاده می‌شود. TRSM به گونه‌ای طراحی شده است که تنها یک دستور فیزیکی را می‌شناسد و در صورتی که فردی برای دستیابی کلیدها تلاش نماید آنها را از بین می‌برد.
- HSM (Hardware Security Module): یک سخت‌افزار رمزنگاری و یا یک کارت توسعه می‌باشد که اکثرا در سیستم‌های پشتی برای مدیریت امن کلیدها و رمزگشایی استفاده می‌شود. HSM قابلیت مدیریت کلیدها بر اساس نیازمندی‌های استانداردهای X9.24-1، TR-39 و PCI HW-

P2PE را فراهم می‌سازد. HSMهای خاصی نیز به عنوان ماشین‌های تزریق کلید جهت تزریق PIN و کلیدهای P2PE در پایانه‌های پرداخت و کارت‌خوان‌ها استفاده می‌شود.

محافظت از داده صاحب کارت

استانداردهای PCI تنها رمزنگاری دیسک ذخیره‌سازی و در برخی موارد رمزنگاری ارتباطات را لازم می‌دانند. از آنجایی که تکنولوژی فرآیند پرداخت کارتی مشکلات امنیتی اساسی دارد، برنامه پرداخت باید داده حساس صاحب کارت را در هنگام استقرار در حافظه، در دیسک ذخیره‌سازی و در هنگام انتقال رمز کند. علاوه بر این، پیاده سازی اصل دفاع در عمق -به کار بردن لایه‌های امنیتی اضافی در هر جایی که امکانپذیر باشد- نیز ایده خوبی می‌باشد. برای مثال، در هنگام ارسال داده در یک شبکه، برنامه پرداخت می‌تواند داده‌های حساس را با استفاده از الگوریتم‌های متقارن رمز کند، و همچنین کل جلسه ارتباطی را با مکانیزم‌های امنیت انتقال مانند SSL، HTTPS و یا IPSec رمزگذاری کند. در تئوری، کنترل‌های امنیتی منطقی و فیزیکی نیز می‌توانند لایه دیگری از حفاظت را شکل دهند ولی در محیط کاری POS که مستقیما در دسترس عموم قرار دارد موثر نیست.

داده مستقر در حافظه

در صورتی که داده حساس صاحب کارت قبل از استقرار در حافظه رمزگذاری نشود نمی‌تواند به طور کامل ایمن باشد. در شرایط فعلی مکانیزم امنیتی قابل اطمینانی که بتواند از جمع‌آوری اطلاعات حافظه جلوگیری کند وجود ندارد. در صورتی که یک مهاجم یا هکر به کامپیوتر میزبان پایانه فروش دسترسی پیدا کند، احتمال افشای داده‌ها بسیار بالا است. دلیل این امر این است که اکثر عملیات روی داده‌های حساس شامل رمزنگاری ، رمزگشایی و مدیریت کلید رمزنگاری در حافظه انجام می‌پذیرد.

می‌توان از راهکارهای پیشگیرانه‌ای برای کاهش افشای اطلاعات استفاده کرد. به صورت دقیق‌تر، کاری انجام می‌دهیم تا مدت زمان اقامت داده‌های رمز نشده در حافظه را کاهش دهیم، بنابراین نرم‌افزارهای جمع‌آوری کننده حافظه با پیچیدگی و حساسیت پایین زمان کافی جهت یافتن داده رمز نشده را نداشته باشند. جهت انجام چنین کاری، برنامه پرداخت لازم است داده‌های حساس را بیشتر اوقات در حافظه به صورت رمز شده نگهداری کند و تنها در زمان نیاز به داده رمز نشده آنها را برای زمان کوتاهی رمزگشایی کند. نکته بسیار مهم این است که داده رمز نشده بعد از استفاده از حافظه پاک شود. برای مثال، بافرهای حافظه (آرایه‌های بایتی) که شامل داده‌های حساس می‌باشند، نباید به عملیات زباله‌روبی سپرده شود (که شاید بعد از زمان بسیار طولانی انجام شود)، و حتما باید توسط روش‌های خاصی قبل از اینکه اشاره‌گر آن بافر گم شود، صفر شوند.

شکی نیست که تنها راه قابل اطمینان برای محافظت از داده در حافظه این است که آن‌ها را به صورت رمز نشده در حافظه نداشته باشیم. تکنولوژی‌های رمزگذاری نقطه به نقطه (P2PE) قابلیت رمز کردن داده‌ها قبل از رسیدن به حافظه کامپیوتر میزبان (داخل پایانه پرداخت و یا دستگاه MSR) و رمزگشایی آن تنها بعد از ترک پایانه فروش (در درگاه پرداخت مرکز داده) را در اختیار ما قرار می‌دهند. حتی رمزگذاری نقطه به نقطه نرم‌افزاری، که در آن داده‌ها در برنامه در حال اجرا بر روی دستگاه POI رمزگذاری می‌شوند، با اینکه دارای آسیب‌پذیری‌هایی می باشد اما میزان محرمانگی بیشتری را به نسبت عدم وجود رمزگذاری نقطه به نقطه در اختیار ما قرار می‌دهد. علاوه بر محافظت داده‌های مستقر در حافظه، رمزنگاری نقطه به نقطه عدم دسترسی به داده‌های حساس در هنگام انتقال و در زمان استقرار بر روی دیسک ذخیره‌سازی را نیز تضمین می‌کند.

داده در هنگام انتقال

از داده‌های حساسی که در حال انتقال بر روی یک شبکه هستند می‌توان به دو روش حفاظت کرد:

۱. رمزنگاری محموله: قسمت‌های حساس مشخص شده توسط روش‌های متقارن یا نامتقارن رمزنگاری می‌شوند
۲. رمزنگاری انتقال: کل اتصال ارتباطی با استفاده از پروتکل‌های امن مانند SSL، HTTPS یا IPSec رمزنگاری می‌شوند

هر دو روش را برای فراهم نمودن محافظت چند سطحی و سخت نمودن کار مهاجم می‌توان باهم ترکیب کرد. به این شکل که در صورتی که یک لایه شکسته شود، لایه دیگری همچنان جهت شکستن وجود دارد.

استفاده از SSL

SSL یک پروتکل امن‌سازی است که از استراق سمع، دستکاری و جعل داده‌ها در یک شبکه جلوگیری می‌کند. استفاده از SSL برای محرمانگی و احراز هویت بسیار موفق بوده است. به نسخه‌های جدید SSL امنیت لایه انتقال یا TLS گفته می‌شود ولی همچنان از هر دو نام اختصاری استفاده می‌شود.

SSL نمونه‌ای از استفاده برنامه پرداخت از الگوریتم‌های رمزگذاری می‌باشد. سمت کاربر SSL (که می‌تواند مرورگر اینترنتی و یا برنامه پایانه فروش باشد) از گواهی سرور[۵۰] برای احراز هویت استفاده می‌کند. گواهی سرور، که توسط کاربر در هنگام انجام فرآیند دست‌دادن[۵۱] بارگیری می‌شود، شامل قسمت عمومی زوج کلید نامتقارن می‌باشد و کلید خصوصی در سمت سرور قرار می‌گیرد. SSL از الگوریتم‌های رمزگذاری کلید-

[۵۰] Server Certificate
[۵۱] Handshake Process

عمومی برای تعویض کلید[۵۲] استفاده می‌کند. زمانی که عملیات تعویض کلید تمام شود، داده انتقالی بین کاربر و سرور با استفاده از الگوریتم‌های متقارن رمزگذاری می‌شود. دلیل استفاده از رمزگذاری متقارن این است که این الگوریتم‌ها کارایی بهتری نسبت به الگوریتم‌های نامتقارن دارند.

استفاده از تونل‌های رمزنگاری شده

بسترها یا تونل‌های رمزنگاری شده جایگزین دیگری برای استفاده از پروتکل‌های ارتباطی امن مانند SSL و HTTPS محسوب می‌شود. مزیت استفاده از تونل رمزنگاری شده –که توسط پروتکل‌های شبکه خصوصی مجازی (VPN) و یا پروتکل‌های تونل‌زنی مانند IPSec- این است که نیازی به ایجاد تغییر در کد و تنظیمات برنامه در هر دو سمت کاربر و سرور نیست. این ویژگی در زمان استفاده از نرم‌افزارهای قدیمی که مکانیزم‌های امنیتی در آنها پیاده‌سازی نشده است و یا از مکانیزم‌های رمزگذاری به صورت پیش‌فرض پشتیبانی نمی‌کنند، مفید واقع می‌شود. نکته منفی پروتکل‌های تونل‌زنی این است که برای استفاده از آنها نیازمند اعمال تنظیمات و یا نصب نرم‌افزار خاصی بر روی سیستم‌های کاربران می‌باشد، کاربرانی که که در اکثر مواقع خارج از کنترل ارائه دهنده برنامه پرداخت هستند.

امنیت پروتکل اینترنت یا IPSec بهترین جایگزین SSL در هنگام کار با کاربران متعدد است. IPSec معمولا نیازی به نصب برنامه‌های اضافی بر روی سیستم کاربران یا سرور ندارد و لازم نیست تغییری در برنامه مورد محافظت در سمت کاربر و یا سرور ایجاد شود. سیستم عامل ویندوز به صورت پیش‌فرض از IPSec پشتیبانی می‌کند و می‌توان با استفاده از تنظیمات شبکه و گواهی‌های مورد نیاز آن را تنظیم و استفاده کرد.

داده ذخیره شده

بهترین راهکار برای مشکل محافظت از داده‌های ذخیره شده، در قدم اول اجتناب از ذخیره داده‌های حساس می‌باشد، که البته در عمل این کار دشواری است. در عمل، موارد بسیاری وجود دارد که برنامه پرداخت باید به صورت موقت داده‌هایی مانند S&R، TOR و ... را ذخیره کند. دومین راهکار مناسب استفاده از رمزگذاری نقطه به نقطه است. در صورتی که به دلایلی استفاده از رمزگذاری نقطه به نقطه نیز امکان‌پذیر نباشد، می‌توان از رمزگذاری مرسوم (معمولا متقارن) استفاده کرد.

یک راهکار مناسب استفاده از دو (یا حتی بیشتر) کلید رمزگذاری می‌باشد که با عنوان‌های کلید رمزگذاری کلید (KEK) و کلید رمزگذاری داده (DEK) شناخته می‌شوند. KEK در هنگام اجرای برنامه از چندین مؤلفه ساخته می‌شود. تنها هدف آن محافظت از DEK است که خود داده‌های حساس را رمزگذاری می‌کند.

[۵۲] Key Exchange

ایده پشت استفاده از KEK پیچیده و سخت‌تر کردن عملیات بازیابی کلید است. مزیت دیگر استفاده از KEK این است که به DEK اجازه می‌دهد به صورت پویا برای هر جلسه رمزگذاری تولید شود.

رمزگذاری نقطه به نقطه

رمزگذاری نقطه به نقطه که محافظت همزمان داده در حافظه، در هنگام ذخیره سازی و انتقال را فراهم می-آورد بسیار قدرتمند بوده و به صورت گسترده مورد استفاده قرار می‌گیرد. این تکنولوژی در حال حاضر پیاده‌سازی شده و یا توسط بسیاری از سازندگان نرم‌افزارهای پرداخت در حال توسعه می‌باشد.

ایده رمزگذاری نقطه به نقطه به شکل کلی ساده است: داده حساس در یک سمت ارتباط رمز می‌شود و در سمت دیگر رمزگشایی می‌شود. با این حال، زمانی که رمزگذاری نقطه به نقطه به ساختار محیط خرید و فروش اعمال می‌شود، چندین شرایط باید مورد توجه قرار گیرد:

- داده باید در نزدیکترین مکان به نقطه ورود رمز شود. در مورد رمزگذاری نقطه به نقطه سخت-افزاری، شیارهای مغناطیسی در داخل TRSM دستگاه MSR رمزگذاری می‌شود.
- سمتی که رمزگذاری می‌کند می‌تواند در یک محیط خطرناک مانند فروشگاه قرار گیرد. بنابراین سمتی که رمزگشایی می‌کند باید در محیط بسیار امن (هم منطقی و هم فیزیکی) مانند HSM که در یک مرکز داده نصب شده است، قرار گیرد.
- کلیدهای رمزگذاری باید با استفاده از فرآیندهای امن خاص و امکاناتی که از افشای آنها جلوگیری می‌کند مدیریت شوند.

راهکارهای رمزگذاری نقطه به نقطه در ساختار مختلفی بسته به تکنولوژی‌های مورد استفاده عرضه می‌شوند که در جدول ۷-۳ آمده است. اول اینکه، تمام راهکارهای رمزگذاری نقطه به نقطه را می‌توان به دو گروه اصلی رمزگذاری نقطه به نقطه سخت‌افزاری (HW-P2PE) و رمزگذاری نقطه به نقطه نرم‌افزاری (SW-P2PE) تقسیم کرد. تفاوت آنها در این است که نوع سخت‌افزاری آن از سخت‌افزار برای عملیات رمزنگاری استفاده می‌کند، در صورتی که نوع نرم‌افزاری آن عملیات رمزگذاری (و در برخی مواقع رمزگشایی) را در نرم‌افزار انجام می‌دهد. نوع سخت‌افزاری بسیار امن‌تر است، به این دلیل که در نوع سخت‌افزاری به دلیل استفاده از سخت‌افزار رمزنگاری (مانند TRSM و HSM) در برابر نفوذ نسبت به نوع نرم‌افزاری بسیار محافظت‌شده‌تر (هم در منطق و هم به صورت فیزیکی) است. با این حال، نوع نرم‌افزاری نیز گزینه مناسبی می‌باشد، حداقل وجود آن بهتر از نبودن هیچگونه رمزگذاری نقطه به نقطه است.

جدول ۷-۳: انواع مختلف تکنولوژی‌های P2PE

نوع P2PE	سمت رمزگذاری	سمت رمزگشایی	استاندارد PCI

رمزگذاری نقطه به نقطه سخت‌افزاری (HW-P2PE)	رمزگذاری و مدیریت کلید توسط سخت‌افزار انجام می‌شود (TRSM)	رمزگشایی و مدیریت کلید توسط سخت‌افزار انجام می‌شود (HSM)	PCI Hardware/ Hardware P2PE (نسخه اولیه در سپتامبر ۲۰۱۱ عرضه شد)
نوع ترکیبی Hardware/Hybrid P2PE	رمزگذاری و مدیریت کلید توسط سخت‌افزار انجام می‌شود (TRSM)	رمزگشایی و مدیریت کلید توسط نرم‌افزار انجام می‌شود	PCI Hardware/ Hybrid P2PE (در دسامبر ۲۰۱۲ عرضه شد)
رمزگذاری نقطه به نقطه نرم-افزاری (SW-P2PE)	رمزگذاری و مدیریت کلید توسط نرم‌افزار انجام می‌شود	رمزگشایی توسط نرم‌افزار انجام می‌شود؛ مدیریت کلید توسط سخت‌افزار انجام می‌شود (HSM)	مشخص نیست

پرداخت‌های موبایلی و بدون تماس

راهکارهای پرداخت مبتنی بر NFC از پایانه‌های پرداخت بدون تماس موجود برای وارد کردن داده کارت به POS استفاده می‌کنند. برای این کار داده‌های کارت را در دستگاه موبایل ذخیره می‌کنند، که البته این داده‌ها می‌توانند به سرقت بروند. علاوه بر این، خواننده‌های خط داده مغناطیسی[۵۳] یا MSD بدون تماس نسبت به MSRهای معمولی امن‌تر نیستند. زمانی که داده از طریق NFC از چیپ کارت (یا فرستنده NFC دستگاه موبایل) به پایانه پرداخت انتقال داده می‌شود، به صورت داخلی توسط POS و برنامه پرداخت دقیقا به همان شکل که داده از طریق MSR معمولی خوانده می‌شود، مدیریت می‌شود.

راهکارهای غیر از NFC می‌توانند موارد ذکر شده را برطرف سازند. چنین راهکارهایی از POS برای ارتباط یک دستگاه موبایل به تراکنش پرداخت استفاده می‌کنند. تمامی داده‌های حساس بین POS و سرور پرداخت موبایل تبادل می‌شود و بنابراین هیچ داده حساسی در سطح ذخیره‌سازی وجود ندارد. ساختار سنتی کارت‌های اعتباری نیز می‌توانند حفظ شوند و هیچ انقلاب تکنولوژیکی در سطح کارت لازم نیست، چون داده‌ها به صورت امن در مراکز داده که تمامی نیازمندی‌های لازم برای حفاظت مناسب را دارا می-باشند، ذخیره شده‌اند.

[۵۳] Magnetic Stripe Data