Я.Ю. Коршунов

**Особенности сертификации эмиссионной части процессинга крупного банка по стандарту PCI DSS**

**Введение**

Как известно, сертификация на соответствие стандарту PCI DSS начинается с определения области применимости стандарта, обычно описываемой перечнем бизнес-процессов. Если говорить про банки, то можно выделить эмиссионные и эквайринговые процессы. Эмиссионные процессы, если говорить упрощенно, то это те, в рамках которых обрабатываются данные карт, выпущенных самим банком. С точки зрения прикладного уровня информационной инфраструктуры, это, например, учетные системы персонализации и выдачи карт клиентам, их логистики, а также АБС, использующие номера карт клиентов при расчетах. Кроме того, к ним могут относиться CRM, использующие номера карт при идентификации клиентов и подобные системы.

К эквайринговым процессам относят те процессы, которые обрабатывают транзакции от банкоматов или POS-терминалов, содержащие данные любых карт, в том числе других банков, входящие запросы от МПС и переводы карта-карта между банками.

Какие процессы нужно включать в область аудита? Давайте попробуем разобраться.

Если банк начал свой карточный бизнес недавно, когда стандарт PCI DSS уже существовал и МПС предъявляли свои требования к процессингу, то у банка есть все шансы построить свои бизнес-процессы в полном соответствии со стандартом. Это возможно, потому что у большинства «коробочных» решений на рынке, как правило, уже есть готовые интерфейсы работы с другими автоматизированными системами, исключающие использование данных платежных карт в открытом виде в интеграциях.

Но если это банк с давней историей и он начал выпускать карты и принимать платежи задолго до возникновения стандарта PCI DSS и появления соответствующих профильных служб и внутренних регламентирующих документов, то все не так просто.

Как правило, в этом случае сильно распространено использование номеров платежных карт, эмитированных самим банком, как уникальных идентификаторов и связок между системами. Это действительно очень удобно, так как соблюдается два принципа: уникальность номера карты и однозначная идентификация владельца. И не нужно придумывать какие либо другие идентификаторы. При этом такая архитектура получила широкое распространение. Проще перечислить старые автоматизированные системы, не использующие номера карт, чем системы, их использующие.

Стандарт PCI DSS непосредственно не делит карты на собственные и сторонние, поэтому распространяется на все бизнес-процессы: и эквайринг, и эмиссию. Но представители МПС, как правило, не настаивают на сертификации эмиссионной части, и QSA-аудиторы следуют такой же логике. При этом банк обязан явным образом взять на себя риски компрометации данных по своим картам. Технически проблем здесь, по большому счету, нет. При допущении компрометации данных своих карт эти карты можно быстро заблокировать, перед клиентами извиниться, компенсировать им возможные потери и на этом считать инцидент исчерпанным. Издержки такого подхода могут перекрываться доходом от карточного бизнеса, и обычно банки принимают на себя такие риски. Конечно, это не значит, что системы, обрабатывающие данные карт своего банка можно никак не защищать. Защищать нужно, и требования стандарта выполнять нужно, но сертифицировать эти системы не обязательно, приняв все риски.

В отношении же данных карт сторонних банков ситуация отличается. Сторонние карты – это чужие активы, которыми рискует банк. Просчитать все последствия компрометации не удастся, так как банк не может предсказать последствия утечки данных, не может никак отследить реальные потери держателей скомпрометированных карт, и для идентификации и блокировки карт вынужден обращаться в сторонние банки, в том числе, и через МПС. Более того, банк обязан сообщать в МПС обо всех фактах таких утечек. Поэтому эквайринговый процесс всегда находится в зоне особого внимания МПС и совета PCI SSC.

**Сертификация эмиссии**

Итак, нужно или не нужно сертифицировать эмиссию? Если нужно, то как избавиться от данных платежных карт в исторических данных и бурно развивающихся бизнес-процессах?

Прежде всего, нужно адекватно оценить необходимость сертификации. Для чего она? Ничего не мешает строить эффективную защиту эмиссионных данных без сертификации. Но сертификация дает несколько преимуществ. Во-первых, это всегда независимая оценка степени защищенности, это большой опыт аудиторов в выявлении узких мест в защите и обмен опытом по повышению эффективности защиты. Во-вторых, это очень удобный внутренний стимул – некий набор требований, которые нужно безусловно выполнить, чтобы соответствовать стандарту, без ослабления механизмов защиты в некоторых, по мнению внутренних сотрудников, не очень важных точках. В третьих, это дает значимый репутационный эффект и возможность маркетинговых заявлений о некоторых преимуществах на рынке за счет наиболее передовой защиты, подтвержденной сертифицированными аудиторами.

Одного желания сертифицировать эмиссию недостаточно. Нужно, чтобы была возможность архитектурных преобразований большого количества систем банка. Для этого нужно заинтересовать широкий круг подразделений. Начать пересматривать бизнес-процессы, вовлеченные в обработку карт, дорабатывать или перенастраивать работу автоматизированных систем, поменять у сотрудников отношение к этой проблеме, внедрить культуру работы с платежными картами, распространить свои требования на всех контрагентов.

Итак, как можно сертифицировать все бизнес-процессы банка по PCI DSS? Вариантов немного. Первый – это «натянуть» требования стандарта на всю инфраструктуру банка. Что это значит? Это значит, что, безусловно, все системы должны попасть в область аудита. Звучит просто, но при реализации есть нюансы. Большое количество разных систем в скоупе будет означать большое количество сотрудников, имеющих легитимный доступ к данным платежных карт, значит, будет возникать большое количество событий безопасности, которые должны регистрироваться и обрабатываться для выявления инцидентов. А это и большие затраты на программно-аппаратные комплексы для их обработки и на сотрудников службы ИБ, которые должны эти данные интерпретировать. Представьте себе тысячу высоконагруженных серверов. Сколько нужно ресурсов, чтобы переварить все, что на этих серверах происходит? Сравнимое по мощности оборудование для построения подсистем ИБ, громадный штат сотрудников ИБ, большое количество событий и высокий риск пропустить инцидент среди них. Конечно, сложные системы фильтрации и корреляции событий способны снизить нагрузку на штат ИБ, но их развитие и поддержание – также непростая задача, в свою очередь, требующая ресурсов. Второй вариант – это отказаться от обработки карточных данных в большинстве автоматизированных систем банка, перейдя на маскированные (правильно – усечённые, но для простоты будем говорить маскированные) данные, если это позволяет бизнес-процесс или некие сквозные идентификаторы, которые однозначно связаны с номерами карт. Если систем больше сотни, и в каждой из них, так или иначе, уже есть какие-то свои внутренние идентификаторы, то процесс внедрения новой интеграционной сущности может оказаться нетривиальным и привести к огромным трудозатратам с неопределенным результатом в конце.

Если следовать лучшим мировым практикам и основам подхода стандарта PCI DSS по формированию области аудита, то мы видим, что список АС, обрабатывающих карточные данные, должен быть максимально ограничен для достижения максимально эффективной защиты данных.

**Где можно обрабатывать номера карт**

Каковы критерии, можно ли системе обрабатывать карточные данные, или нельзя? Исходя из смысла самого номера карты, можно предположить, что система тогда имеет возможность работы с номерами карт, когда этот номер ей необходим именно как оригинальная последовательность цифр, имеющая бизнес-значение. В основном, это осуществление транзакции, при которой цифры номера карты непосредственно участвуют в формировании финансового сообщения в международные платежные системы. Транзакции, а точнее, авторизация и клиринг осуществляются в основной процессинговой системе. Также можно выделить системы по претензионной работе с клиентами, где есть сценарии проверки каждой цифры номера карты при осуществлении переводов клиентами банка (например, при переводе произошла ошибка или опечатка и необходимо сравнить введенный номер карты с оригиналом у клиента по обращению в поддержку). Системы фрод-мониторинга, где критичным сервисом является скорость работы, для минимизации задержки в обслуживании клиентов можно строить на основе не самих номеров карт, а их преобразованных значений, например, различных хешей. Но, зачастую, даже такая задержка, требуемая для вычисления хеш-значений, может не устраивать бизнес и системы фрод-мониторинга вынуждены работать с полными номерами карт. Отдельным пунктом нужно выделить работу с обращениями государственных органов – судебные запросы, запросы от МВД, финансовый мониторинг, судебные приставы и т. п. Традиционно государственные регуляторы требуют предоставления информации по данным карт. Совет PCI SSC здесь имеет гибкую позицию: все локальные нормативные акты и законы имеют приоритет над международными требованиями. Вернее, стандарт PCI DSS не замещает локальное законодательств, если дословно. В Российских законах ничего не сказано про защиту данных при передаче информации, но, как правило, существуют договора по обмену информацией, где защита передаваемых данных требуется в явном виде. Поэтому никакого конфликта между PCI DSS и локальными законами нет, они друг другу не противоречат, а наоборот, дополняют. И, если сама справка для запрашивающего органа, действительно, может быть составлена с содержанием полных номеров карт, и даже отправлена заинтересованным лицам по открытым каналам, например, по электронной почте, это не значит, что документ не нужно защищать при передаче и хранении.

Мы ограничили область аудита. Теперь нужно решить другую важную задачу: чем заменить номера карт при их обработке в автоматизированных системах? Какие будут к этому заменителю требования? Это зависит от архитектурного принципа перехода. Основной вопрос: это будет дополнительный атрибут карты или замена самого номера на другое значение? В обоих случаях есть свои плюсы и минусы. Введение дополнительного атрибута позволит построить последовательность работы по переходу, растянуть процесс перехода во времени и обеспечить гибкость при интеграционных доработках, потому что каждая система может продолжать какое-то время работать как с номером карты, так и с его заменителем. Однако с точки зрения безопасности такой вариант имеет большую проблему. Это накопление локальных копий связок номеров карты с новым идентификатором. Существование таких кешей повышает риск компрометации этих данных и, впоследствии, использования в мошеннических целях внутри банка. Причем отследить это достаточно сложно, так как это будет сопряжено с детальным исследованием кода и структуры базы данных приложения. Если не вводить дополнительный атрибут, а заменить номер карты на новую сущность, то вышеуказанная проблема не стоит так остро. Строго говоря, в момент замены также можно допустить накопление локальных кешей, но это отследить и предотвратить существенно проще, так как операция локализована местом и временем. Правда, при замене номера карты на новую сущность нужно обеспечить единовременную замену во всех интегрированных системах, что существенно повышает требования к синхронности процедур миграции для предотвращения расхождения данных. Так что, каким путем идти, нужно решать коллегиально с привлечением архитектуры, ИТ, ИБ и бизнеса.

**Токенизация**

Давайте назовем нашу новую сущность именем «токен» для компактности. Попробуем сформировать для него критерии. Итак, чем должен являться токен? Случайной величиной, неким хешем или иной криптографической производной от номера карты? Какова должна быть его длина? Допустимы ли в нем буквы или шестнадцатеричные цифры? Использование каких-либо хешей не подходит, так как в системах в любом случае будет сохраняться усеченные (маскированные) номера карт и подбор эти хешей – технически возможен и сводит на ноль всю идею токенизации. Иные криптопреобразования по сути также являются вычисляемыми в случае компрометации ключа, чего исключить полностью также нельзя. Получается, что случайная величина будет наиболее правильной базой формирования токена. При этом, естественно, необходимо не допустить коллизий, то есть совпадений значений токенов для разных исходных номеров карт. Это легко реализовать, если случайной величиной будет не само значение токена, а смещение в таблице криптопреобразований от исходных данных. При этом криптопреобразование гарантирует нам отсутствие коллизий, а случайное смещение (или «перемешивание») данных делает невозможным подбор путем прямого перебора.

Какова должна быть длина токена и его атрибутный состав? Понятно, что чем длиннее значение токена, тем надежнее и устойчивее будет работать система, но есть один нюанс. Если мы говорим о замене номеров карт в существующих системах, то мы должны учитывать и ограничения, применяемые к исходным данным. В общем случае можно считать, что наиболее простым способом замены будет замена на величину, соразмерную исходным данным. То есть, 16-значный номер карты должен быть заменен на 16-значный токен. При таком подходе доработки в системах в целом не будут выглядеть очень сложными. Два важных момента. Первый: токен должен быть легко отличим от номера карты и гарантировано не представлять собой номер другой карты. Выполнить эти два условия очень просто: необходимо гарантировать наличие хотя бы одной шестнадцатеричной цифры A-F в любом токене, тогда и любые алгоритмы на лету смогут понять, номер карты это или токен и уж точно токен не совпадет ни с каким другим номером любой карты. Второй момент: так как системы будут отображать в интерфейсах маскированный номер, то они либо будут хранить эту сущность либо каждый раз при выводе проводить детокенизацию (получение исходного номера карты из токена) с одновременным маскированием. Давайте совместим усеченный номер карты и токен в одном значении, так как, все равно, маскированный номер необходим для работы интерфейсов. Можно ли это сделать? Если мы сохраним, например, первые шесть и последние четыре цифры номера, то оставшиеся шесть нужно заменить на токенную часть. Будет ли этого достаточно? Давайте посчитаем емкость этих данных. Для 16-значного номера карты (наиболее популярный) маскируется 6 цифр, и именно они должны являться базой для токена, так как остальные остаются неизменными. 6 цифр – это миллион значений. А для шестнадцатеричных цифр – это 16,8 миллионов. Вычтем один миллион значений, не содержащих шестнадцатеричных цифр A-F (000 000 – 999 999), остается 15,8 миллионов значений. Используя первые шесть и последние четыре цифры в качестве значения для вычисления сдвига по таблице замены (по модулю 15), можно реализовать токенизацию с уникальными токенными значениями для всех токенизируемых шести цифр с пятнадцатикратным запасом, случайно распределенным по всем совпадающим токенизированным данным.



Рисунок 1. Токен и номер карты. Сходства и различия.

Что является узким местом в данной конструкции? Получается, что только лишь сама таблица замены токенов, то есть необходимо обеспечить локализацию таблицы соответствия токенизированных данных и ее безопасное хранение и использование. Для этого необходимо исключить постоянное накопление таблицы соответствия где бы то ни было, кроме самой системы токенизации.

Где проводить преобразование номера карты в токен? Очевидно, что это необходимо делать на границе сетевых зон безопасности контура PCI DSS и остальной сети, если централизованно автоматизировать замену номера карты на токен при интеграциях. Если предполагать, что замена будет не централизованная, а выполняемая непосредственно в АС, то возникнет сложность синхронизации разработок использования токенизации в разных командах разработчиков, поэтому централизация в данном случае надёжнее и потратит меньше ресурсов. При этом нагрузка на систему преобразования и доступность системы должна быть спроектирована с запасом на пиковые нагрузки системы в наиболее транзакционно ёмкие периоды года и учесть миграционный период, когда одновременно много автоматизированных систем начнут конвертацию своих исторически накопленных данных.

**Миграция данных**

Процедура миграции с номеров карт на токены также требует проработки. В условиях больших банков осуществить синхронную доработку, а тем более, миграцию накопленных данных очень сложно. Поэтому важно предусмотреть поэтапную миграцию. Этапы могут быть как в разрезе автоматизированных систем, так и в разрезе номеров карт, например, по диапазонам BIN. А можно реализовать и комплексный подход. Выбор варианта нужно делать, учитывая все архитектурные особенности инфраструктуры, готовность всех участников к синхронизации и приоритезации своих планов. Как правило, неизбежно будут возникать задачи, реализация которых носит временный характер, а после окончания миграции результаты по этим задачам станут неактуальны. Например, если две автоматизированных системы дорабатываются, но в разные сроки, то при сохранении интеграционных связей между ними нужно учитывать необходимость создания временного механизма захода на токенизацию и детокенизацию данных. Таких архитектурных «временных» решений будет много. Это нужно понимать, и предварительно обсуждать со всеми, не снижая важности промежуточных мероприятий, которые, в конечном итоге, и определят успех всей кампании.

По окончании миграции перед сертификацией необходимо убедиться, что во всех исторических данных, логах, архивах, копиях все карточные данные также мигрированы или уничтожены, очень важно это не забыть проверить.



Рисунок 2. Использование токенизации на границе контура PCI DSS и остальным банком.

**Опыт Сбербанка**

В Сбербанке в конце 2019 года стартовала кампания по токенизации всех автоматизированных систем, не входивших ранее в область аудита PCI DSS, но обрабатывающих карточные данные (по собственным картам) для того, чтобы в перспективе выйти на сертификацию по стандарту PCI DSS в части эмиссии. Подготовка к этому велась несколько последних лет. В рамках новой линейки продуктов Банка и выхода новой платформы реализации приложений выведен в промышленную эксплуатацию модуль токенизации. Кроме этого, подготовлены архитектурные планы перехода, разработаны и синхронизированы детальные планы доработки приложений и планы миграции данных. Самым сложным предстоит пересмотр бизнес-процессов, работающих с карточными данными, так как масштабы Сбербанка такие, что любое изменение должно сопровождаться колоссальной работой, административными и организационными мероприятиями. Но цель поставлена, руководство заинтересовано – значит, изменения, в любом случае, произойдут.

**Заключение**

Итак, поставив себе амбициозную цель сертификации эмиссии в крупных банках, нужно быть готовым к очень серьезному объему работы. Начиная с аналитики, планирования, пилотирования, заканчивая комплексным тестированием и длинным процессом миграции данных. Только совместная работа бизнес-подразделений, администраторов, архитекторов, безопасности, разработки, тестирования и многих других под началом высшего руководства банка может привести к успешному результату.