Сервистестированиякорректностинастройки на SSLQUALYS SSL LABS - SSL SERVER TEST

джомбо яшо

17 июня 2015 г.

1 Ход работы

1.1 Изучение

Изучить лучшие практики по развертыванию SSL/TLS Конфигурация

- Используйте безопасные протоколы

Существует пять версий протоколов в SSL/TLS семейства: SSL v2, SSL v3, TLS v1.0, TLS v1.1 и TLS v1.2. Из них: • SSL v2 является небезопасным и не должен быть использован. • SSL v3 является небезопасным при использовании с HTTP и слабым при использовании с другими протоколами. Эта версия также устарела, поэтому она не должна использоваться. • TLS v1.0 до сих пор является безопасным протоколом. При использовании с HTTP этот протокол обеспечивает безопасность, но только при тщательной конфигурации. • TLS v1.1 и v1.2 не имеют известных проблем безопасности.

TLS v1.2 должен быть вашим основным протоколом. Эта версия лучше, потому что она поддерживает важные функции, которые недоступны в более ранних версиях. Если ваш сервер (или любое промежуточное устройство) не поддерживает TLS v1.2, то планируйте его модернизацию в ускоренном режиме. Если ваши поставщики услуг не поддерживают TLS v1.2, требуйте, чтобы они модернизировали свою систему.

Для поддержки более старых клиентов вы должны продолжать поддерживать TLS v1.0 и TLS v1.1 еще некоторое время. С некоторыми обходными путями эти протоколы еще можно считать достаточно безопасными для большинства веб-сайтов.

- Используйте безопасные алгоритмы шифрования

Для безопасного обмена данными вы должны сначала убедиться, что вы общаетесь непосредственно с нужным абонентом (и не через кого-то, кто будет подслушивать). В SSL и TLS алгоритмы шифрования используются для определения, насколько безопасно происходит обмен данными. Они состоят из различных строительных блоков. Если в одном из строительных блоков наблюдается слабая безопасность, то вы должны быть в состоянии переключиться на другой. Ваша цель — использовать только те алгоритмы шифрования, которые обеспечивают аутентификацию и шифрование в 128 бит или более. Всего остального следует избегать: • Наборы

со слабыми алгоритмами шифрования (как правило, от 40 до 56 бит) могут быть легко взломаны • RC4 также сейчас считается слабым. Вы должны убрать поддержку этого алгоритма как можно раньше, но только после проверки потенциального негативного воздействия на совместимость. • 3DES обеспечивает около 112 бит безопасности. Это ниже рекомендованного минимума 128 бит, но это все еще достаточно сильный алгоритм. Большая практическая проблема в том, что 3DES гораздо медленнее, чем альтернативные варианты. Таким образом, мы не рекомендуем его для повышения производительности.

- Контроль за выбором алгоритма шифрования

В SSL версии 3 и более поздних версиях протокола, клиенты отправляют список алгоритмов шифрования, которые они поддерживают, и сервер выбирает один из них для организации безопасного канала связи. Не все сервера могут делать это хорошо, так как некоторые выбирают первый поддерживаемый алгоритм из списка. Таким образом, выбор правильного алгоритма шифрования является критически важным для безопасности.

- Поддержка Forward Secrecy.

Forward Secrecy — это особенность протокола, который обеспечивает безопасный обмен данными, он не зависит от закрытого ключа сервера. С алгоритмами шифрования, которые не поддерживают Forward Secrecy, возможно расшифровать ранее зашифрованные разговоры с помощью закрытого ключа сервера. Нужно поддерживать и предпочитать ECDHE (аббревиатура ECDHE расшифровывается как «эфемерный алгоритм Диффи-Хеллмана с использованием эллиптических кривых») алгоритмы шифрования. Для поддержки более широкого круга клиентов, вы должны также использовать DHE, как запасной вариант после ECDHE.

- Отключите Renegotiation по инициативе клиента

В SSL / TLS renegotiation позволяет сторонам остановить обмен данными, с тем чтобы повторно инициировать его для обеспечения безопасности. Есть некоторые случаи, в которых renegotiation должен быть инициирован сервером, но нет никакой известной необходимости позволять инициировать renegotiation клиентом. Кроме того это может облегчить организацию DDoS-атаки на ваши сервера.

- Снижение известных проблем

В какой-то момент могут возникнуть проблемы с безопасностью с любым продуктом. Хорошо, если вы всегда в курсе событий в мире информационной безопасности. По крайней мере, вы должны следить за релизами безопасности продуктов, которые используете, и устанавливать их, как только они становятся доступными.

Следите за тем, что происходит в мире безопасности и адаптируйтесь к ситуации, когда это необходимо. По крайней мере, вы должны сразу устанавливать патчи, закрывающие обнаруженные уязвимости, как только они становятся доступными. Обратите внимание на следующие вопросы:

— Отключите TLS compression

В 2012 году CRIME attack показал, как TLS сжатие может быть использовано злоумышленниками для выявления деталей конфиденциальных данных (например, сессионные куки). Очень немногие клиенты поддерживали TLS сжатие тогда (и в настоящее время), так что маловероятно, что вы будете испытывать какие-либо проблемы с производительностью после отключения TLS сжатия на серверах.

— Отключите RC4

Алгоритм RC4 является небезопасным и должен быть отключен. В настоящее время мы знаем, что для взлома RC4 требуются миллионы запросов, много пропускной способности и времени. Таким образом, риск все еще относительно невелик, но вполне возможно, что атаки будут масштабнее в будущем. Перед снятием RC4 проверьте, будут ли ваши существующие пользователи затронуты; другими словами, проверить, если у вас есть клиенты, которые поддерживают только RC4.

— Отключить SSL v3

SSL v3 уязвим против POODLE атаки, которая была обнаружена в октябре 2014. Лучший способ устранения уязвимости POODLE атаки — это отключить SSL v3, который в большинстве сайтов можно сделать безопасно.

1.2 Изучить основные уязвимости и атаки на SSL последнего времени – POODLE, HeartBleed

2 Практическое задание

2.1 Интерпретировать результаты в разделе Summary Выбрать со стартовой страницы SSL Server Test один домен из списка Recent Best и один домен из списка Recent Worst–изучитьотчеты,интерпретироватьрезультатывразделеSummary получаем : (SSL Report: www2.mizunousa.com (50.205.43.98))

2.2 расшифровать все аббревиатуры шифров в раделе Configuration

TLS – Transport Layer Security.

SSL – Secure Sockets Layer.

RSA – аббревиатура от фамилий Rivest, Shamir и Adleman.

RC4 – **R**ivest **c**ipher 4 или **R**on's **c**ode 4.

SHA – Secure Hash Algorithm.

AES – Advanced Encryption Standard.

CBC – Cipher Block Chaining.

3DES – Triple Data Encryption Standard.

SNI – Server Name Indication

NPN – Next Protocol Negotiation.

HSTS – HTTP Strict Transport Security.

HPKP – HTTP Public Key Pinning.

HTTP – HyperText Transfer Protocol.

2.3 большинство позиций в разделе Protocol Details

Secure Renegotiation – возобновление подключения TLS.

BEAST attack – атака утилитой BEAST (Browser Exploit Against SSL/TLS).

POODLE – уязвимость, позволяющая расшифровать содержимое защищённого канала коммуникации.

Downgrade attack – атака, при которой пользователя вынуждают использовать менее безопасные протоколы, которые всё ещё поддерживаются из соображений совместимости.

TLS compression – В 2012 году CRIME attack показал, как TLS сжатие может быть использовано злоумышленниками для выявления деталей конфиденциальных данных (например, сессионные куки).

Heartbleed – ошибка в OpenSSL, позволяющая несанкционированно читать память на сервере до 64 килобайт за один запрос. Атаку можно производить бесконечное количество раз.

Forward Secrecy – особенность протокола, который обеспечивает безопасный обмен данными, он не зависит от закрытого ключа сервера. С алгоритмами шифрования, которые не поддерживают Forward Secrecy, возможно расшифровать ранее зашифрованные разговоры с помощью закрытого ключа сервера.

Next Protocol Negotiation – клиент сообщает серверу по каким протоколам он бы хотел общаться и сервер может ответить наиболее предпочтительным из тех, которые он знает.

Strict Transport Security – механизм, активирующий форсированное защищённое соединение по HTTPS. Данная политика безопасности позволяет сразу же устанавливать безопасное соединение, вместо использования HTTP. Механизм использует особый заголовок HTTP Strict-Transport-Security, для переключения пользователя, зашедшего по HTTP, на HTTPS-сервер.

2.1 Выводы

Сервер использует сертифисат и зашишен от следущих атак : heart-blead. Данный сертификат защищает пустую страницу, Успешная атака BEAST похожа на взлом сессии. К сожалению, для смягчения угрозы со стороны сервера требуется RC4, который больше не рекомендуется. Из-за этого, а также из-за того что атака BEAST теперь в значительной степени уменьшается на стороне клиента, мы больше не рекомендуем смягчения на сервере путем использования RC4. В некоторых ситуациях, когда есть большое количество старых клиентов, уязвимых для атаки BEAST, более безопасно использовать RC4 с TLS 1.0 и более ранние версии протокола. Принимать это решение следует осторожно и только после полного понимания окружающей среды и модели ее угроз.