

Лабораторная работа №7
Курс: Защита информации

Воробьев Олег

8 июня 2015 г.

Содержание

1	Цель работы	3
2	Ход работы	3
2.1	Изучить описания десяти самых распространенных веб-уязвимости согласно рейтингу OWASP.	3
2.2	HTTP Splitting	5
2.3	Недостатки контроля доступа	6
2.4	Безопасность AJAX	6
2.5	Недостатки аутентификации	6
2.6	Переполнение буфера	7
2.7	Качество кода	7
2.8	Многопоточность	7
2.9	Межсайтовое выполнение сценариев	8
2.10	Неправильная обработка ошибок	8
2.11	Недостатки приводящие к осуществлению инъекций (SQL и прочее)	9
2.12	Отказ в обслуживании	10
2.13	Небезопасное сетевое взаимодействие	11
2.14	Небезопасная конфигурация	11
2.15	Небезопасное хранилище	11
2.16	Исполнение злонамеренного кода	11
2.17	Подделка параметров	12
2.18	Недостатки управление сессией	12
3	Вывод	12

1 Цель работы

Изучить сервис тестирования корректности настройки SSL на сервере Qualys SSL Labs и его основные возможности.

2 Ход работы

2.1 Изучение

Изучить лучшие практики по развертыванию SSL/TLS

- Использовать 2048-битные закрытые ключи.

Используйте 2048-битный RSA или 256-битные ECDSA закрытые ключи для всех ваших серверов. Ключи такой крепости безопасны и будут оставаться безопасными в течение значительного периода времени.

- Защитить закрытый ключ.

Относитесь к закрытым ключам как к важным активам, предоставляя доступ к как можно меньшей группе сотрудников.

- Обеспечить охват всех используемых доменных имен.

Убедитесь, что ваши сертификаты охватывают все доменные имена, которые вы хотите использовать на сайте.

- Приобретать сертификаты у надежного CA центра.
- Использовать надежные алгоритмы подписи сертификата.

Безопасность сертификата зависит от длины закрытого ключа и прочности используемой функции хеширования. Сегодня большинство сертификатов используют алгоритм SHA1, который считается слабым.

- Использовать безопасные протоколы. (TLS v1.0/v1.1/v1.2)
- Использовать безопасные алгоритмы шифрования.

В данном случае подойдут симметричные алгоритмы с ключами более 128 бит.

- Контролировать выбор 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.

- Использование 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.

- Отключить проверку защищенности по инициативе клиента.

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

POODLE Атака POODLE работает по следующему сценарию: Взломщик отправляет свои данные на сервер по протоколу SSL3 от имени вменяемой структуры, что позволяет ему постепенно расшифровывать данные из запросов. Это возможно, так как в SSL3 нету привязки к MAC адресу.

HeartBleed Эта атака использует уязвимость уязвимость криптографии OpenSSL, позволяя несанкционированно читать память на сервере и на клиенте, в том числе и для извлечения закрытого ключа сервера. Иллюстрация ниже наглядно продемонстрирует как осуществляется данная атака.

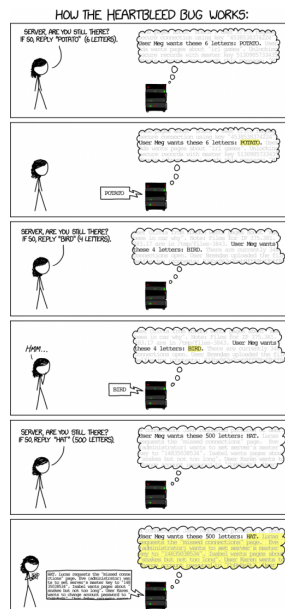


Рис. 1: Атака HeartBleed.

2.2 Практическое задание

Выбрать со стартовой страницы SSL Server Test один домен из списка Recent Best и один домен из списка Recent Worst - изучить отчеты, интерпретировать результаты в разделе Summary.

Recent Best SSL Report: roughgrain.com (87.117.231.91)

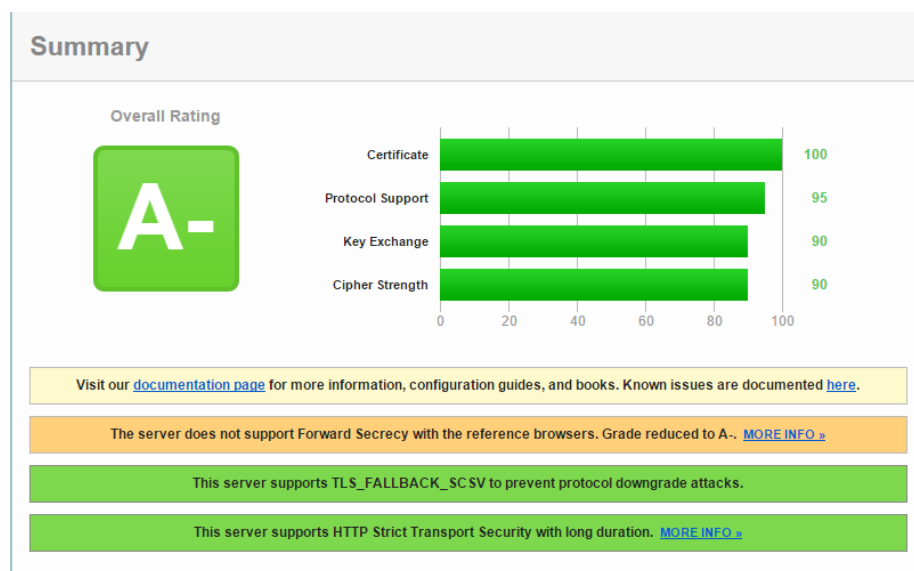


Рис. 2: Recently best.

- Поддерживает все типы протоколов TLS
- Не поддерживается Forward Secutity для старых браузеров
- Поддерживает длительное форсированное защищенное соединение через HTTPS
- Защищен от downgrade атак

Recent Worst SSL Report: kbase.creator.zoho.com (54.251.251.126)

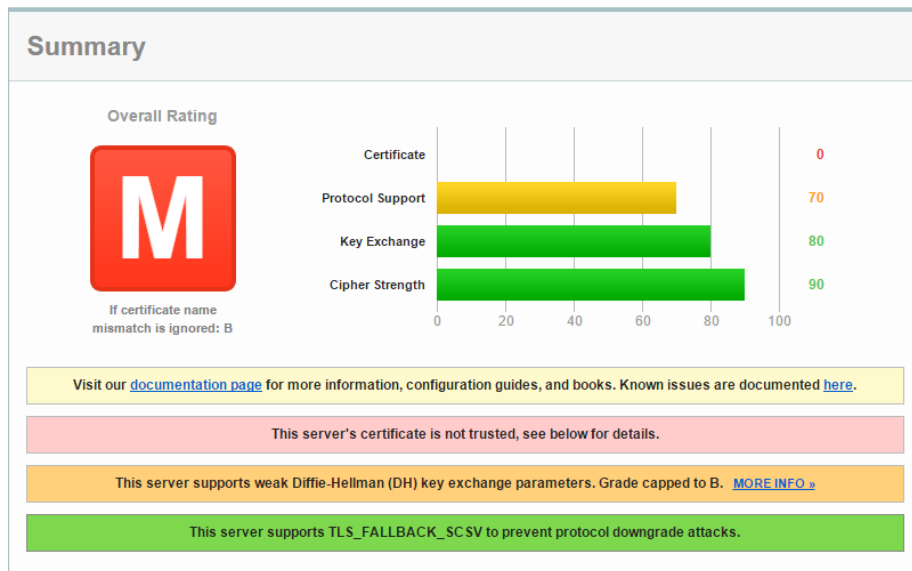


Рис. 3: Recently worst.

- Сертификат не заверен
- Используется слабый алгоритм Диффи Хельмана
- Защищен от downgrade атак
- Не поддерживается Forward Sequtity для старых браузеров

Самостоятельный анализ Для анализа был выбран сервер строителя приложений zoho (SSL Report: zohoplatform.com (74.201.154.174))

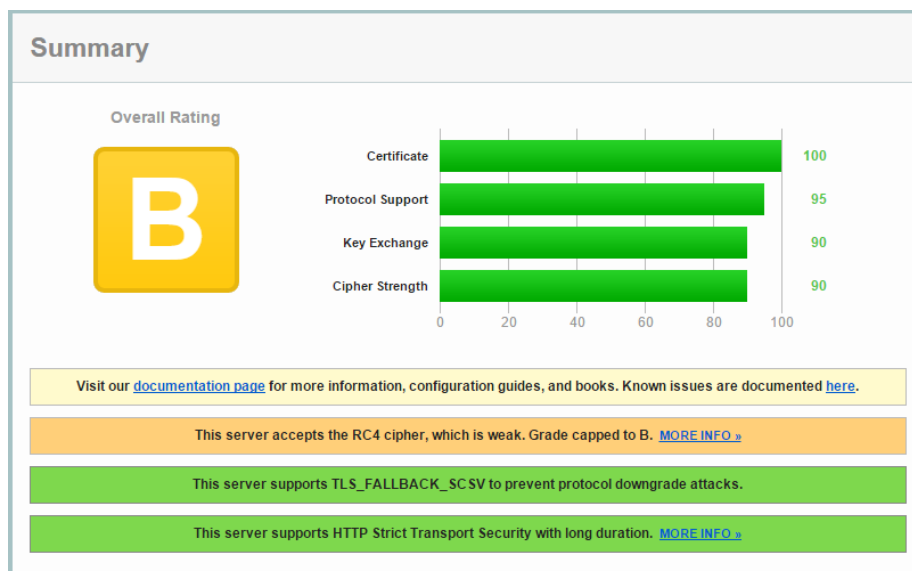


Рис. 4: Zoho сервер.

Интерпретировать результат в разделе summary

- Используется слабый алгоритм шифрования RC4
- Защищен от downgrade атак
- Поддерживает длительное форсированное защищенное соединение через HTTPS

Расшифровать все аббревиатуры шифров в разделе Configuration
Вот список используемых алгоритмов.

Cipher Suites (SSL 3+ suites in server-preferred order; deprecated and SSL 2 suites always at the end)			
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)	ECDH 256 bits (eq. 3072 bits RSA)	FS	128
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)	ECDH 256 bits (eq. 3072 bits RSA)	FS	256
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (0xc027)	ECDH 256 bits (eq. 3072 bits RSA)	FS	128
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (0xc028)	ECDH 256 bits (eq. 3072 bits RSA)	FS	256
TLS_RSA_WITH_AES_128_GCM_SHA256 (0x9c)			128
TLS_RSA_WITH_AES_256_GCM_SHA384 (0x9d)			256
TLS_RSA_WITH_AES_128_CBC_SHA256 (0x3c)			128
TLS_RSA_WITH_AES_256_CBC_SHA256 (0x3d)			256
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc013)	ECDH 256 bits (eq. 3072 bits RSA)	FS	128
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)	ECDH 256 bits (eq. 3072 bits RSA)	FS	256
TLS_RSA_WITH_AES_256_CBC_SHA (0x35)			256
TLS_RSA_WITH_AES_128_CBC_SHA (0x2f)			128
TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA (0xc012)	ECDH 256 bits (eq. 3072 bits RSA)	FS	112
TLS_RSA_WITH_3DES_EDE_CBC_SHA (0xa)			112
TLS_RSA_WITH_RC4_128_SHA (0x5)		WEAK	128
TLS_ECDHE_RSA_WITH_RC4_128_SHA (0xc011)		WEAK	128

Рис. 5: Список алоритмов.

Расшифруем некоторые аббревиатуры.

- TLS ECDHE означает Алгоритм Диффи-Хэлмана на эллиптических кривых
- RSA это алгоритм шифрования с открытым ключом.
- AES 128 так же алгоритм шифрования с длиной ключа в 128 бит
- GCM и CBC это два режима блочного шифрования
- SHA256 -это хэш функция с длиной клбча в 256 бит

Прокомментировать большинство позиций в разделе Protocol Details Вот что выдает нам сервис.

Protocol Details	
Secure Renegotiation	Supported
Secure Client-Initiated Renegotiation	No
Insecure Client-Initiated Renegotiation	No
BEAST attack	Not mitigated server-side (more info) TLS 1.0: 0xc013
POODLE (SSLv3)	No, SSL 3 not supported (more info)
POODLE (TLS)	No (more info)
Downgrade attack prevention	Yes, TLS_FALLBACK_SCSV supported (more info)
TLS compression	No
RC4	Yes WEAK (more info)
Heartbeat (extension)	Yes
Heartbleed (vulnerability)	No (more info)
OpenSSL CCS vuln. (CVE-2014-0224)	No (more info)
Forward Secrecy	With modern browsers (more info)
Next Protocol Negotiation (NPN)	Yes http/1.1
Session resumption (caching)	Yes
Session resumption (tickets)	Yes
OCSP stapling	No
Strict Transport Security (HSTS)	Yes max-age=15768000
Public Key Pinning (HPKP)	No
Long handshake intolerance	No
TLS extension intolerance	No
TLS version intolerance	No
Incorrect SNI alerts	-
Uses common DH prime	No
SSL 2 handshake compatibility	Yes

Рис. 6: Protocol details.

- Строки 1-3 Перепроверка сертификата и защищенность этого процесса
- Строки 4-7 Уязвимости к атакам Poodle, Beast, downgrade
- Строка 9 Используется слабый алгоритм RC4
- Строка 10 -11 уязвимость HeartBleed
- Строка 13 Совместимость Forward Secrecy с браузерами

- Строка 14 Наличие NPN
в настоящее время используется для согласования использования SPDY в качестве протокола прикладного уровня на порт 443, а также для выполнения SPDY согласования версии.
Основной задачей SPDY является снижение времени загрузки веб-страниц и их элементов. Это достигается за счёт расстановки приоритетов и мультиплексирования передачи нескольких файлов таким образом, чтобы требовалось только одно соединение для каждого клиента.
- Строка 15-16 Параметры сессии
- Строка 18 Реализация HSTS
HSTS — механизм, активирующий форсированное защищённое соединение через протокол HTTPS. Данная политика безопасности позволяет сразу же устанавливать безопасное соединение, вместо использования HTTP-протокола. Механизм использует особый заголовок HTTP Strict-Transport-Security, для переключения пользователя, зашедшего по HTTP, на HTTPS-сервер.
- Строка 19 Реализация HPKP
Позволяет указать, какой сертификат выдан доверительным центром, а какой нет, это позволяет отклонить TLS соединения с сайтов, CA которых заведомо неправильный. Это мешает использовать такие атаки как "человек посередине".
Функция связывает набор хэшей открытых ключей для доменного имени, например: при подключении к сайту, используя TLS браузера, гарантирует, что есть пересечение открытых ключей в компьютерной цепочке доверия и множества отпечатков, связанных с этим доменом. Эта проверка выполняется во время верификации сертификата фазы связи, до того, как данные посылаются или обрабатываются браузером.
- Строка 25 Совместимость с SSL2 handshake

Сделать итоговый вывод о реализации SSL на заданном домене
В целом, если судить по работе сервиса, то конфигурация сервера весьма неплохая. Сервер использует доверенный сертификат и защищен от некоторых типов атак. Однако до сих пор используется алгоритм RC4, что является проблемой. Forward Secrecy реализовано не для всех браузеров, однако реализации отсутствуют только для самых старых, так что это можно не относить к первичным проблемам. В целом, сервер имеет необходимый набор защиты, при желании может быть взломан, поэтому в будущем стоит обратить внимание на усиление некоторых параметров, начать можно с отключения протокола RC4.

3 Вывод

В ходе данной работы были изучены возможности сервиса Qualys SSL LABS. Данный сервис анализирует качество реализации защиты домена, а так же предоставляет отчет об используемых технологиях и об известных уязвимостях сервера. Также можно посмотреть используемые протоколы. Стоит отметить, что использование такого рода сервисов важно в коммерческом плане, но только для первичного анализа.