Методы аутентификации, Secure Remote Password

http://www.cypherpunks.ru/

Многофакторная аутентификация

Сильнейшая аутентификация (вспомним фильмы про ядерное оружие) представляет комбинацию следующих методов:

Что вы знаете — пароли, парольные фразы

Что вы имеете — аппаратные токены (криптоключи), приватные асимметричные ключи, одноразовые пароли

Что вы есть — биометрика

Пароли

- Простой софт
- Мало ресурсов для обработки
- Не нужны третьи лица
- Не нужно ничего дополнительного аппаратного

Пароли

- Простой софт
- Мало ресурсов для обработки
- Не нужны третьи лица
- Не нужно ничего дополнительного аппаратного

- Сложно придумать, сложнее запомнить (мало энтропии) $\Rightarrow$ мало толку на практике
- Нельзя передавать по нешифрованным каналам $\Rightarrow$ уже надо обменяться ключом и установить шифрованную связь
- Компрометация сервера приводит у утере секрета $\Rightarrow$ для каждого ресурса свой пароль
- Уязвимы к словарным атакам

Пароли

Парольные фразы

Длинный «пароль» из низкоэнтропийных данных (не очень то и случайные слова).

Пароли

Парольные фразы

Длинный «пароль» из низкоэнтропийных данных (не очень то и случайные слова).

- Тривиальная реализация на уровне софта
- Легко придумать, легко запомнить
- Много энтропии

Появился толк!

Пароли

Парольные фразы

Длинный «пароль» из низкоэнтропийных данных (не очень то и случайные слова).

- Тривиальная реализация на уровне софта
- Легко придумать, легко запомнить
- Много энтропии

Появился толк!

Остаются программисты делающие отвратительные интерфейсы

Пароли

Усиление

PBKDF2, bcrypt, scrypt ...

Пароли

Усиление

PBKDF2, bcrypt, scrypt ...

- Не сложно реализовать в софте
- Компрометация БД сервера не раскрывает секреты

Пароли

Усиление

PBKDF2, bcrypt, scrypt ...

- Не сложно реализовать в софте
- Компрометация БД сервера не раскрывает секреты

- Ресурсоёмкость
- Всё ещё уязвимы к словарным атакам
- Не совместимо с CHAP

Пароли

CHAP

Передача пароля не в открытом виде, а в виде хэша пароля и заранее полученного случайного *challenge* от сервера.

Пароли
CHAP

Передача пароля не в открытом виде, а в виде хэша пароля и заранее полученного случайного *challenge* от сервера.

- Можно использовать с нешифрованными каналами

Пароли
CHAP

Передача пароля не в открытом виде, а в виде хэша пароля и заранее полученного случайного *challenge* от сервера.

- Можно использовать с нешифрованными каналами
- Необходимо изменение протокола аутентификации

Пароли

Менеджеры паролей

Пароли придумываются и хранятся специализированной программой локально у пользователя. БД защищена парольной фразой.

Пароли

Менеджеры паролей

Пароли придумываются и хранятся специализированной программой локально у пользователя. БД защищена парольной фразой.

- Совместимость с ужасными программистами
- Один сервис $\rightarrow$ один пароль
- Хорошая энтропия

Пароли

Менеджеры паролей

Пароли придумываются и хранятся специализированной программой локально у пользователя. БД защищена парольной фразой.

- Совместимость с ужасными программистами
- Один сервис $\rightarrow$ один пароль
- Хорошая энтропия

- Дополнительный софт у пользователя
- Одна точка отказа

Сервисы одноразового входа

OAuth1, OAuth2, OpenID ...

Сервисы одноразового входа

OAuth1, OAuth2, OpenID ...

- Отсутствие дополнительного софта у пользователя

Сервисы одноразового входа

OAuth1, OAuth2, OpenID ...

- Отсутствие дополнительного софта у пользователя

- Одна **неподконтрольная** точка отказа
- Создание инфраструктуры на сервере для общения с сервисами
- Утечка приватной информации третьим лицам

Клиентские сертификаты

Публичные асимметричные ключи, используемые для проверки подписи приватным. Это уже не знание, а имение.
Зашифрованы локально у пользователя парольной фразой.

Клиентские сертификаты

Публичные асимметричные ключи, используемые для проверки подписи приватным. Это уже не знание, а имение.
Зашифрованы локально у пользователя парольной фразой.

- Много софта где это доступно
- Нет секретов человека $\Rightarrow$ компрометация сервера не так страшна
- Один и тот же сертификат можно использовать на многих сервисах

Клиентские сертификаты

Публичные асимметричные ключи, используемые для проверки подписи приватным. Это уже не знание, а имение.
Зашифрованы локально у пользователя парольной фразой.

- Много софта где это доступно
- Нет секретов человека $\Rightarrow$ компрометация сервера не так страшна
- Один и тот же сертификат можно использовать на многих сервисах

- Сложность реализации
- Недостаточная грамотность пользователей для безопасного применения
- Нагрузка на сервер
- Одна точка отказа, неустойчивая к словарным атакам
- Уязвимость к атакам на криптографические примитивы

Клиентские сертификаты

Привязка каналов (channel binding)

Установка долгоживущего шифрованного соединения, за счёт кэширования сессионных ключей или тикетов-доступа (в контексте TLS).

Клиентские сертификаты

Привязка каналов (channel binding)

Установка долгоживущего шифрованного соединения, за счёт кэширования сессионных ключей или тикетов-доступа (в контексте TLS).

- Сильная разгрузка сервера
- Аутентификация проходит один раз и надолго

Клиентские сертификаты

Привязка каналов (channel binding)

Установка долгоживущего шифрованного соединения, за счёт кэширования сессионных ключей или тикетов-доступа (в контексте TLS).

- Сильная разгрузка сервера
- Аутентификация проходит один раз и надолго

- Необходимость (пусть и небольшого) хранилища сессионных ключей, либо тикетов-доступа

Клиентские сертификаты

Аппаратные криптоключи, TPM

Перенос хранилища приватного асимметричного ключа и функций его использующих на отдельные компьютеры с защищённой от вскрытия памятью. Доступ ограничен цифровым кодом, как правило, с ограниченным количеством попыток.

Клиентские сертификаты

Аппаратные криптоключи, TPM

Перенос хранилища приватного асимметричного ключа и функций его использующих на отдельные компьютеры с защищённой от вскрытия памятью. Доступ ограничен цифровым кодом, как правило, с ограниченным количеством попыток.

- Неприменимость словарных атак

Клиентские сертификаты

Аппаратные криптоключи, TPM

Перенос хранилища приватного асимметричного ключа и функций его использующих на отдельные компьютеры с защищённой от вскрытия памятью. Доступ ограничен цифровым кодом, как правило, с ограниченным количеством попыток.

- Неприменимость словарных атак

- Дополнительное дорогое аппаратное обеспечение пользователю
- **Крайне** сложно убедиться в добротной хорошей реализации криптоключа. По факту чуть ли не большинство сделано неприемлемо криптографически слабо

Одноразовые пароли

Часто это аппаратные токены со встроенной простой функцией MAC-а, выдающие зависимые синхронизированные с сервером короткие цифровые ключи на дисплее.

Одноразовые пароли

Часто это аппаратные токены со встроенной простой функцией MAC-а, выдающие зависимые синхронизированные с сервером короткие цифровые ключи на дисплее.

- Простота реализации программного а аппаратного обеспечения
- Отсутствие медленной асимметричной криптографии

Одноразовые пароли

Часто это аппаратные токены со встроенной простой функцией MAC-а, выдающие зависимые синхронизированные с сервером короткие цифровые ключи на дисплее.

- Простота реализации программного а аппаратного обеспечения
- Отсутствие медленной асимметричной криптографии

- Дополнительное аппаратное обеспечение
- Для каждого сервиса отдельный токен
- Разсинхронизация сулит много головной боли

Сильные протоколы аутентификации

Password Authenticated Key Exchange

Интерактивные протоколы использующие принцип нулевого неразглашения (zero–knowledge password proof) и генерирующие сессионные ключи шифрования и аутентификации заодно.
EKE, SPEKE, DH-EKE ... ...

Сильные протоколы аутентификации

Password Authenticated Key Exchange

Интерактивные протоколы использующие принцип нулевого неразглашения (zero–knowledge password proof) и генерирующие сессионные ключи шифрования и аутентификации заодно.
EKE, SPEKE, DH-EKE ... ...

- Zero–knowledge гарантирует бесполезность перехвата трафика (атака на криптографические низлежащие примитивы неприменимы)
- SPEKE гарантирует ещё и совершенную прямую секретность

Сильные протоколы аутентификации

Password Authenticated Key Exchange

Интерактивные протоколы использующие принцип нулевого неразглашения (zero–knowledge password proof) и генерирующие сессионные ключи шифрования и аутентификации заодно.
EKE, SPEKE, DH-EKE ... ...

- Zero–knowledge гарантирует бесполезность перехвата трафика (атака на криптографические низлежащие примитивы неприменимы)
- SPEKE гарантирует ещё и совершенную прямую секретность

- Пароль должен знать и сервер тоже
- Изменение протоколов

Сильные протоколы аутентификации

Augmented PAKE

PAKE протоколы в которых сервер не знает самого секрета. Хранится всего-лишь некий проверщик (verifier). A–SPEKE, B–SPEKE, SRP ...

- Сервер не знает пользовательского секрета $\Rightarrow$ атака на сервер тоже становится бесполезной

Сильные протоколы аутентификации

Secure Remote Password

Не использует шифрование как таковое. Использует в основе хорошо проверенные временем схожие с DH алгоритмы.

- Гораздо быстрее RSA, DSA, ElGamal, DH, RSA
- При компрометации одной криптографической функции безопасность всё–ранво остаётся

Сильные протоколы аутентификации

Secure Remote Password

- zero–knowledge проверка секрета
- бесполезность перехвата трафика и атак на криптографические примитивы
- совершенную прямую секретность
- высокая производительность
- одновременное с аутентификацией установление сессионных ключей шифрования и аутентификации (трафика)
- независимость от третьих сервисов
- юридическая/патентная чистота
- относительная простота реализации

Резюме

- Вместо паролей весь хороший софт использует парольные фразы уже давно
- А хороший серверный софт усиливает пароли при хранении и не передаёт в открытом виде
- Хорошая однофакторная аутентификация лучше чем слабые и не удобные двух–факторные: правильная парольная фраза надёжнее многих других средств
- Сильные протоколы аутентификации, как и основанные на сертификатах защищённых парольной фразой, имеются много в каком софте, а также на уровне протоколов типа TLS
- SRP проверен временем, легко внедряется даже на уровне библиотеки JavaScript

Спасибо за внимание!

Вопросы?