Week 6 readings notes

Jonathan Lam

10/14/21

Contents

1 Diffie-Hellman key exchange

(author?) (year?)

- (didn't take notes on the first half b/c of time constraints; basics of DHE)

- Diffie-Hellman problem: Given an element g and the values of g^x and g^y, what is the value of g^{xy}?
- No long-term private keys are held – a and b can be discarded at the end of a session
- The original DH exchange by itself doesn't provide authentication and is subject to a MITM attack
  - STS protocol is a variant of DH exchange that prevents MITM

1.1 Other uses

- Password-authenticated key agreement (PAKE)
- Public key: relatively simple, same parts are exposed to the attacker
  - The public key is now $(g^a \mod p, g, p)$, private key is a; $g^b issentintheopen$ along with the encrypted message
  - Not used in practice because the RSA algorithm is used; it has a certificate authority
  - However, many of the same algorithms are related to it
- Cryptocurrency

2 Imperfect forward secrecy: how Diffie-Hellman fails in practice

Adrian et al., 2015.

- Logjam proposed: TLS flaw that lets a MITM downgrade connections to "export-grade" DH; allows breaking of 512-bit codes after week-long precomputation
- DH is the main key exchange mechanism in SSH and IPsec and is popular in TLS
- **"Export-grade" cryptography** (and **export ciphers**): see `https://crypto.stackexchange.com/a/41772`; some software was crippled to use weaker cryptography as part of the "crypto wars"; if this code is still present it can be exploited as Logjam does

- Current best technique for attacking DH is to compromise private component by computing discrete log of public values
- With precomputation for the group for a prime p, arbitrary discrete logs can be calculated quickly in that group
- Logjam is an attack on the TLS protocol as opposed to an implementation flaw like FREAK
- Also able to exploit other vulnerabilities outside of TLS: bad math (composite-order subgroups and short exponents, inability of clients to properly validate DH parameters without knowing the subgroup order)
- Suggests that the NSA may be exploiting 1024-bit DH to decrypt VPN traffic
- Suggestions for mitigation: only allow larger DH groups, disable export-grade cryptography; migrate to stronger DH groups (e.g., based on elliptic curves)
- Some standardized strong ("safe") primes; these are good but if widely reused they can be attacked in this manner
  - The number one 512-bit prime is very commonly used, accounting for over 90% of servers that use 512-bit primes
- Snowden was right! (probably)

2.1 Active attack implementation

- MITM attack sitting between TLS client and any server supporting `DHS_EXPORT` and using the most common 512-bit prime (the one used in Apache)
- Steps:
  1. Downgrades connection towards server
  2. Computes session keys
  3. Takes over connection towards client by impersonating the server
- Challenge: compute shared secret g^{ab} before handshake completes in order to forge a `Finished` message from the server

  - To mitigate this, we can use non-browser clients, or TLS warning alerts to prolong the timeout
  - To make the user not notice it, the request can be made on some background resource that the user will not notice
  - Servers also sometimes reuse b (ephemeral key caching)
  - TLS false start: an extension to reduce connection latency by not waiting for the `Finished` message to arrive

2.2 Other exploitable security issues

- 512-bit primes in non-export DHE: don't even have to have an active role, can passively decrypt
- Composite-order subgroups (non-safe primes, i.e., $(p-1)/2$ is composite) and short-order primes
- Groups misconfigured with DSA parameters

2.3 Is NSA breaking 1024-bit DH?

- Look at IKE, key establishment protocol used in IPsec VPNs
- System called TURMOIL which passively collects and decrypts VPN traffic; maintains a database called CORALREEF
- Systems do not always choose the strongest DH key in IKE, even when stronger groups are offered

3 New directions in cryptography

Diffie and Hellman, 1976

- "The best known cryptographic problem is that of privacy: preventing the unauthorized extraction of information from communications over an insecure channel"
- Two approaches to transmitting keying information over public (insecure) channels without compromising the security of the system: both eliminate the need of a **secure key distribution channel**
  - **Public key cryptosystem**: distinct keys, E and D, s.t. computing D from E is computationally infeasible; E is the public key and placed in a public directory

        * Multiple-access cipher
    - **Public key distribution systems**: requires a communication between two users to derive a shared public key

- Note that authentication is a separate, second problem
    - Currently the validity of contracts is guaranteed by signatures
    - **One-way authentication problem**
    - **One-time pads** are the only non-cryptographically broken method, but they require extremely long keys (which are expensive to produce)

3.1 Conventional cryptography

- "**Cryptogrraphy** is the study of 'mathematical' systems for solving two kinds of security problems: privacy and authentication"
- "A channel is considered **public** if its security is inadequate for the needs of its users"
    - Thus public/private of a channel depends on the needs of the users
    - Channels may be threatened by injection a/o eavesdropping
- Divide authentication into **message authentication** and **user authentication**
- Use **secure channels** to distribute keys; but these do not have the bandwidth or lack of latency that a regular channel provides
- "A **cryptographic system** is a single parameter family $|S_K|_{K\in\{K\}}$ of invertible transformations $S_K : \{P\} \to \{C\}$ from a space of plaintext messages to a space of ciphertext messages. The parameter is called the key and is selected from a finite set called the keyspace. If the message spaces are equal, we will denote them both by $\{M\}$. When discussing individual cryptographic transformations S_K, we will sometimes omit mention of the system and merely refer to the transformation K."
- Security guarantees:
    - **Computationally secure**: secure due to computational cost of cryptanalysis, but which would succumb to an attack with unlimited computation

    - Contains sufficient information to uniquely determine the plaintext and the key; security resides in the cost of computing them
  - **Unconditionally secure**: secure no matter how much computation is thrown at it; belongs to the field of Shannon theory of information theory
    - Results from the existence of multiple meaningful solutions to a cryptogram
    - E.g., one-time pads

- Cryptographic systems can be divided into two broad classes
  - **Stream ciphers**: process the plaintext in small chunks (bits or characters)
  - **Block ciphers**: act in a purely combinatorial fashion on large blocks of text, in such a way that a small change in the input block produces a major change in the resulting output (**error propagation** property)
    - This paper deals primarily with block ciphers

- A cryptographic system intended to guarantee privacy will not, in general, prevent replay attacks

- To guarantee authenticity of a message, information is added which is a function not only of the message and a secret key, but of the date and time as well, so that only one who has the key can generate a message with the proper date and time

- Error propagation property is important to prevent against fuzz attacks (which will corrupt much more and be detected as inauthentic)

- Possible threats:
  - **Ciphertext only attack**: cryptanalyst possesses only cyphertext; weakest, most common, attacks that succeed of this kind means that system is totally insecure
  - **Known plaintext attack**: cryptanalyst possesses a substantial quantity of corresponding plaintext and ciphertext; prevents users from releasing past information

  - **Chosen plaintext attack**: cryptanalyst can submit an unlimited number of plaintext messages and examine the resulting cryptograms, often called **IFF** (identify friend or foe)
  - Assumed that cryptanalyst knows the system $\{S_K\}$

- The goal in cryptography is to build systems which are difficult, rather than easy, to identify (**system identification problem**)

- Also worry about the repudiation problem: **threat of dispute**: prevent the receiver from producing seemingly authentic messages

3.2 Public key cryptography

- Cannot assume that users will have made cryptographic preparations to talk to everyone else (e.g., sharing keys between every possible pair of users)

- Public key cryptosystem is a pair of familiars E_K and D_K of algorithms representing invertible transformations on a finite message space M, such that:

  1. For every key K, E_K is the inverse of D_K
  2. For every key and message, the algorithms E_K and D_K are easy to compute
  3. For almost every key, each esily computed algorithm equivalent to D_K is computationally infeasible to derive from E_K
  4. For every key, it is feasible to compute inverse pairs E_K and D_K from K

- "Essentially what is required is a one-way compiler: one which takes an easily understood program written in a high level language and translates it into an incomprehensible program in some machine language"

3.3 Public key distribution system

- Merkle: **public key distribution system**: allow two users to securely exchange a key over an insecure channel; however high transmssion overhead

- New method proposed: good cost ratio that is exponential w.r.t. n, and can authenticate via public read-only channel (basically DFE)

- Makes use of the apparent difficulty of computing logarithms over a finite field $GF(q)$
- Exchanged public key is the $g^a \mod p$ from first paper (however, this promotes long term use of a key)

3.4 One-way authentication

- Need digital phenomenon with the same properties as a written signature: it must be easy for anyone to recognize s authentic, but impossible for anyone other than the legitimate signer to produce it – we call this **one-way authentication**
- One-way functions are commonly used for hashing passwords
- Function is not invertible from computational point of view, but different than (orthogonal to) mathematical non-invertibility
  - Actually, we want to be careful that it is not too mathematically noninvertible, as in the degenerate case when all values hash to the same value any value will generate the correct hash; a small degree of degeneracy is tolerable and common
- Public key cryptosystem can be used to produce a true one-way authentication system as follows: a user can send a message deciphered using the private key, which can be assured of its authenticity by enciphering with the public key

3.5 Problem interrelations and trap doors

- A cryptosystem which is secure against a known plaintext attack can be used to produce a one-way function
  - The converse is sometimes true: it is possible for a function originally found in the search for one-way functions to yield a good cryptosystem
  - A key generator is similar to a one-time pad but uses a pseudorandom number generator with an agreed upon seed (the key)
  - Need to be careful when the one-way function is not uniquely invertible
- A public key cryptosystem can be used to generate a one-way authentication system

  - Converse is not true, making the construction of a public key cryptosystem a strictly more difficult problem than one-way authentication
  - A public key system is really a set of **trap-door one-way functions**: functions which are not really one-way in that simply computed inverses exist. But given an algorithm for the forward function it is computationally infeasible to find a simply computed inverse

- **Trap doors** are more general than trap-door one-way functions: easy to compute in one direction, yet difficult to compute in the opposite direction without special information (called the trap door)
  - A **trap-door cipher** is one which strongly resists cryptanalysis by anyone not in possession of **trap-door information** used in the design of the cipher; there is little evidence for the existence of these currently but it is something to keep in mind
  - "The situation is precisely analogous to a combination lock. Anyone who knows the combination can do in seconds what even a skilled locksmith would require hours to accomplish. And yet, if he forgets the combination, he has no advantage"

- A trap-door cryptosystem can be used to produce a public key distribution system
  - Private key/public key pair is created with the trap-door information, private key holder keeps the trapdoor information
  - By definition, we will require that a trp-door problem be one in which it is computationally feasible to devise the trap door
  - Define a **quasi one-way function** to be not one-way but computationally infeasible even for the designer to find the easily computed inverse; can be used in place of a one-way function
    * Losing the trap-door information to a trap-door one-way function makes it into a quasi one-way function

3.6 Computational complexity

- Mathematical proofs for security of a system fell out of favor for a while until Shannon

- Two major disciplines for studying costs: computational complexity theory and the analysis of algorithms
- The cryptoanalytic difficulty of a system whose encryption and decryptieon operations can be done in P time cannot be greater than NP

4 Questions:

- "It should be difficult for Alice to solve for Bob's private key" – DH p.3
- Extracting key from the shared group element? – DH p.2
- Math part in security section – DH p.4
- PAKE: how does this work? DH p.4
- How does cryptocurrency work? DH p.4
- Ephemeral DH? IFS p.3
- Trap-door ciphers existence? DH74 p.652

5 Review

- STRIDE: Spoofing, Tampering, Repudiablity, Information disclosure, Denial of service, Elevation of privilege