Spring 2020 Cryptography and Network Security

Homework 2

Release Date: 2020/4/14

Due Date: 2020/5/11, 23:59

Instruction

- **Submission Guide:** Please submit all your codes and report to NTU COOL. You need to put all of them in a folder named by your student id, compress it to hw2_{student_id}.zip. For example, hw2_r04922456.zip. The report must be in **PDF** format, and named report.pdf.
- You may encounter new concepts that haven't been taught in class, and thus you're encouraged to discuss with your classmates, search online, ask TAs, etc. However, you must write your own answer and code. Violation of this policy leads to serious consequence.
- You may need to write programs in the Capture The Flag (CTF) problems. Since you can use any programming language you like, we will use a pseudo extension code**.ext** (e.g., code.py, code.c) when referring to the file name in the problem descriptions.
- This homework set are worthy of 160 points.
- You are recommended to provide a brief usage of your code in **readme.txt** (e.g., how to compile, if needed, and execute). You may lose points if TAs can't run your code.
- In each of the Capture The Flag (CTF) problems, you need to find out a flag, which is in `CNS{...}` format, to prove that you have succeeded in solving the problem.
- Besides the flag, you also need to submit the code you used and a short write-up in the report to get full points. The code should be named **code{problem_number}.ext**. For example, code3.py.
- In some CTF problems, you need to connect to a given service to get the flag. These services only allow connections from 140.112.0.0/16, 140.118.0.0/16 and 140.122.0.0/16.

Questions

1. SSL/TLS and Tor (25%)

1. (5%) What is the POODLE attack? How to protect yourself (a client) from this attack?
2. SSL Stripping attack and Super Cookie

a) (5%) What is the SSL Stripping attack? Explain how HTTP Strict Transport Security (HSTS) defends against the SSL Stripping attack.

b) (5%) It seems like HSTS can store one bit of information (whether to use HTTPS or not) in a clients browser. Can you leverage this to create a super cookie to track a user even when the user is browsing your website?
Note 1: *Super Cookie* means you can store a piece of data on the client side even in private web browsing modes.
Note 2: *Track User* means you are able to link a user across multiple browsing sessions on the same website even in private web browsing modes.

c) (5%) How to prevent this kind of attack?

3. About Tor

a) (2%) What is a Tor Entry Guard Relay and how does it work?

b) (3%) What is a Tor Bridge and how does it work?

2. BGP (20%)

In this problem, we would like you to explain and analyze some attacks against the BGP routing protocol. In both attacks, the attacker has control over AS999 and wants to attack AS1000. Figure 1 shows the routing paths in the normal state after AS1000 has announced 10.10.220.0/22. Each circle represents an Autonomous System (AS). A solid line indicates a link over which two neighboring ASes can exchange control messages such as BGP update messages. A dashed line indicates an established AS path to 10.10.220.0/22.

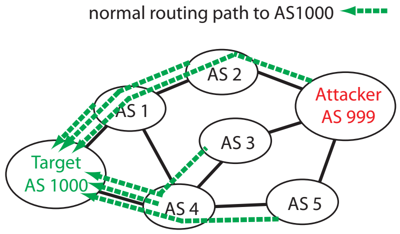


Figure 1: Normal scenario.

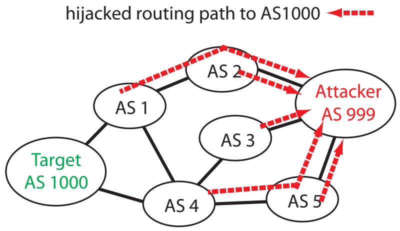


Figure 2: BGP hijacking.

1) (5%) Describe the most likely scenario that could explain the result of Figure 2. Specifically, what did AS999 announce?

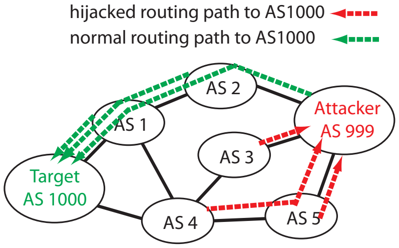

hijacked routing path to AS1000
normal routing path to AS1000
AS 2
AS 1
Attacker
AS 999
AS 3
Target
AS 1000
AS 4
AS 5

Figure 3: BGP man-in-the-middle.

2) In the second type of attack illustrated in Figure 3, the attacker can silently redirect the hijacked traffic back to the victim along the path indicated by green lines. This attack exploits **AS Path Prepending**, where an AS inserts AS numbers at the beginning of an AS path to make this path less preferable for traffic engineering purpose, and **Loop Prevention**, where AS x drops any BGP update with itself (i.e., AS x) in the AS-Path attribute to prevent routing loops.

   a) (5%) Instead of announcing the ownership of an address block, AS999 announces a spurious BGP update. Specify a BGP update message (in the form of {IP prefix, {AS x, AS y, $\cdots$}}) that could cause the result of Figure 3.

   b) (5%) Briefly explain how the attacker misuses path prepending and loop prevention for malicious purpose.

3) (5%)To mitigate BGP prefix hijacking, some propose the idea of BGP Maximum Prefix Limit (MPL). If a prefix length of an advertisement is longer than the MPL, the advertisement will be discarded. For example, if MPL is set to 24 in a BGP speaker, the announcement of 10.10.220.0/25 will simply be ignored. List one advantage and one disadvantage of MPL.

3. Mix Network (18%)

A mix nework is a technique designed to prevent attackers who are able to observe all traffic between nodes from knowing the identity of a packet's sender and receiver. However, in some cases, attackers can somehow retrieve users' relationship. Assume a simple scenario, S is a group of senders, R is a group of receivers and there is a mix between senders and receivers. The mix always buffers b packets from b distinct senders ($b \leq |S|$), padding packets into the same size and then send to receivers in random orders. In other words, a subset S' of all senders S sends b packets to a subset R' of all receivers R in each round, where $|S'| = b$.

(1) (10%) You are an attacker who can observe all flows between end devices and the mix. Assume that $b = 3, S = \{s_1, s_2, ..., s_5\}, R = \{r_1, r_2, ..., r_5\}$, you have collected all flows related to s_1, please find out r_i's identity, (1) is s_1's recipient, (2) is not s_1's

recipient, (3) is uncertain. You should explain how you get the answer in details in your report. The collected data is included in `hw2/mix`.

(2) (8%) Now you are a network developer, please propose a solution against this attack, notice that the availability of your solution should be considered.

4. STS Protocol (20%)

In this problem, we introduce the station to station (STS) protocol, which is a key establishment protocol. The protocol runs as follows:

$$\begin{array}{ccc} P & & Q \\ & \xrightarrow{\qquad\qquad a = g^x \qquad\qquad} & \\ & \xleftarrow{\qquad b = g^y,\ E_k(\sigma_Q(a,b)),\ Cert_Q \qquad} & \\ & \xrightarrow{\qquad E_k(\sigma_P(a,b)),\ Cert_P \qquad} & \end{array}$$

Notation:

- p denotes a cyclic group with a generator g.
- x, y denote random numbers generated by P and Q respectively from the allowed set.
- k denotes the shared secret key $k = H(g^{xy})$, where H is a secure cryptographic hash function.
- $E_k(m)$ denotes secure symmetric encryption of message m using the shared secret key k.
- $\sigma_X(m)$ denotes signature of message m using the private key of user X.
- $Cert_X$ denotes a public key certifcate of user X that can be used to verify the authenticity of X's public key.

STS Protocol begins with one party P generating a random number x and sending g^x to another party Q. Q generates another random number y and computes the shared secret key $k = H(g^{xy})$. Q then signs a message using his private key with the value of g^x and g^y concatenating together, and encrypts the signature using the shared secret key k. P will receive g^y, $E_k(\sigma_Q(a, b))$, and $Cert_Q$. P computes the shared secret key k and decrypts $E_k(\sigma_Q(a, b))$ and verifies the signature using the given $Cert_Q$. P will terminate the protocol if the signature is invalid. Then P will sign the same message g^x and g^y using his private key, encrypt the signature using the shared secret key k, and send it to Q with $Cert_P$. Q will decrypt the message and verify the signature with $Cert_P$.

(1) (5%) Explain why the signatures $\sigma_P(a, b)$, $\sigma_Q(a, b)$ should be encrypted to prevent an *identity misbinding attack*, where P and Q share a same session key, and P thinks he is communicating with Q, but Q thinks he is communicating with adversary. Note

that in an identity misbinding attack, the adversary doesn't have to learn the value of k.

(2) (5%) Suppose P and Q use this protocol and establish a session key k to encrypt messages, and suppose an adversary can force either user to encrypt any message. Show that adversary can perform similar attack as (1) even when the signatures are encrypted.

(3) (5%) Suppose we fix the protocol just as in (2), show another identity misbinding attack in this protocol.

(4) (5%) Consider a variant of STS protocol that runs as follow:

$$
\begin{array}{ccc}
P & & Q \\
 & \xrightarrow{\qquad a = g^x \qquad} & \\
 & \xleftarrow{\quad b = g^y,\ E_k(\sigma_Q(b)),\ Cert_Q \quad} & \\
 & \xrightarrow{\qquad E_k(\sigma_P(a)),\ Cert_P \qquad} &
\end{array}
$$

Show that an adversary can impersonate one of the parties if the protocol uses the RSA signature scheme.

Hint: Suppose the hash function in the RSA signature scheme is an identity function.

Capture The Flag

5. Timmy & Amy (20%)

Hi, my name is Timmy, I have been looking for my friend, Amy, for a long time. Recently, I came up with a method to help me find Amy automatically. What I need to do now is waiting for Amy's response. Ha! Easy peasy. All sources are included in `hw2/timmy`.

(1) (10%) (FLAG1) My friend, Amy, is the only person who knows my favorite number in the world, so I built a system with special verification, only allowing the one who knows the secret number. The server can be accessed by `nc cns.csie.org 10224`.

Note: Timmy's favorite number is a combination of `[0-9]`*.*

(2) (10%) (FLAG2) Wait...you are not Amy! Okay, maybe I should use a more reliable way to find her. I built another system with a mutual authentication protocol, we can use the secret key we shared before to authenticate each other. The server can be accessed by `nc cns.csie.org 10225`.

6. TLS (25%)

You are a secret agent and you are asked to impersonate a member of BALSN, which is a mysterious organization, in order to find out the secret they are hiding. Your colleague has captured some of the communication between two BALSN members. Unfortunately, they are using TLS to protect their messages. Can you crack the protocol and accomplish your mission?

You are provided with a packet capture file, and you can load it up with Wireshark. The file is included in `hw2/tls`.

You can also communicate with the server by connecting `cns.csie.org:10223`. However, please note that it will check your identity in the TLS handshake process.

(1) (10%) (FLAG1) There is a secret in the captured packets. Can you figure it out?

(2) (5%) Why you can decrypt some of the messages in the captured packets, but not all of them?

(3) (10%) (FLAG2) Can you impersonate a BALSN member and communicate with the server?

Hint1: What happen if the two prime factors p and q of an RSA modulus n is too close to each other?

Hint2: The organization name is BALSN.

7. Key Exchange with KDC (32%)

A secure communication channel is very important.

We have learned a key exchange protocol in which a Key Distribution Center (KDC) distributes a shared session key to two users so that both user can use this shared session key to encrypt their messages. The advantage of this protocol is that even if you accidentally leak one of your session keys, your other encrypted messages using different session keys will not be compromised.

To try out this system, you can register an account via `nc cns.csie.org 10220`.

You can also communicate with other users such as Alice (`nc cns.csie.org 10221`) and Bob (`nc cns.csie.org 10222`).

All the files are in `hw2/kdc`.

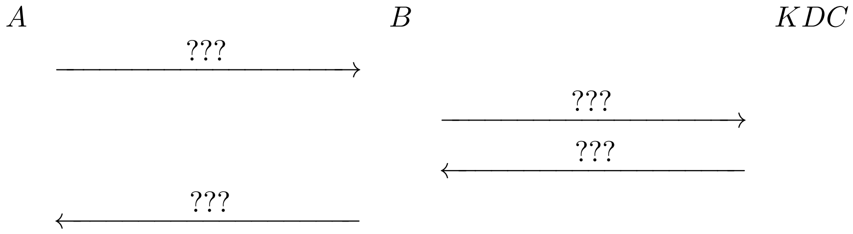
Answer all the following questions in your report:

(1) (4%) Suppose Alice initiates a communication with Bob, can you illustrate a diagram of this protocol by using **ONLY** the following notations:

- A, B and KDC represent Alice, Bob and KDC, respectively.
- N_A and N_B denote random nonce created by A and B, respectively.
- id_A and id_B denote the identities of the users A and B, respectively.
- $E_A(m)$ and $E_B(m)$ denote encryption of message m using shared encryption secret key between KDC and the user, respectively.

- $M_A(m)$ and $M_B(m)$ denote MAC of the message m using shared MAC secret key between KDC and the user, respectively.
- k denotes the shared session key.

Your diagram of this protocol should look like this:



(2) (10%) (FLAG1) There is a flaw in this protocol, can you explain what the flaw is? Then, use this vulnerability to decrypt Alice's encrypted message that should be sent to Bob.

(3) (8%) Can you fix this protocol by modifying the protocol as little as possible? You won't get any credit if the modified protocol is still vulnerable.

(4) (10%) (FLAG2) Can you login as an administrator? Explain how you manage to do that in details.

Important Note1: You need to use different methods to get FLAG1 and FLAG2, otherwise you will only get partial credits.

Important Note2: The server will remove all the users' keys every 120 seconds.