Oana-Alexandra COICA Group 410, Security and Applied Logic
Unsolved exercises: -

Exam

1. True or False

(a) False. 0x5468617473206D79204B756E67204675 can be a valid **temporal key** for WPA/TKIP.

(b) True

(c) True

(d) False. A direct purpose of the Packet Number in the CCMP header is to protect against **replay** attacks.

(e) True

(f) False. 2**26**014829537140 can be an IMSI issued by a Romanian mobile operator.

(g) False. In GSM, there are $\mathbf{2^{128}}$ possible distinct permanent subscriber keys.

(h) True

(i) False. Both the UE and the eNodeB know the value of $\mathbf{K_{eNB}}$ in a successful run of AKA.

(j) False. Public-key cryptography is used in 5G to **encrypt** the response of the UE to the authentication request in 5G-AKA.

2. WPA2 Security

(a)

The Pairwise Master Key (PMK) is the most important element of the key hierarchy, playing the role of the root secret. Different for each mobile device (in case we're talking about a server-key approach), the PMK is not directly used in the encryption process, but plays a major role in the generation of the Pairwise Transient Key (PTK), a collection of separate keys used to secure the link between the supplicant (the mobile device) and the access point:

- EAPOL-Key Encryption Key;

- EAPOL-Key Integrity Key;

- Data Encryption/Integrity Key.

The leakage of the PMK would lead to the leakage of the PTK (which is computed using PMK, AP's MAC, supplicant's MAC and the two nonces that are transmitted in plaintext), making the "secure" channel completely insecure. Old recorded communication sessions (with that device, in case of a server-key approach) would also be compromised, and the future communication sessions (with that device, in case of a server-key approach) would be affected as well until the PMK would be renewed.

(b)

The EAPOLMICKey integrity key is used to prevent anyone from modifying a message without detection. Its leakage offers the following possibilities:

1. Modifying message 2 in the 4-way handshake

a) changing the SNonce => will cause the authenticator to generate a wrong PTK => the handshake will fail;

b) changing the Unicast address => doesn't make sense;

2. Modifying message 3 in the 4-way handshake

a) changing the Install PTK bit => depending on the implementation, the handshake fails (because the supplicant might expect the Install PTK) OR the supplicant is never able to authenticate (Denial of Service);

b) changing the Unicast address => doesn't make sense;

c) changing the Encrypted GTK => doesn't make sense, if the encryption key is not known, the attacker doesn't have control over the GTK itself.

3. Modifying message 4 in the 4-way handshake

a) changing the Unicast address => doesn't make sense;

b) adding an error bit => the authenticator doesn't install the key => the supplicant will figure out later that the authenticator doesn't understand its encrypted packets => the authentication process will be restarted.

Apart from the highlighted possibilities, the compromise of the EAPOLMICKey doesn’t have any other effects on other keys.

(c)

If the AP has a bug that affects its randomization capabilities, reducing the entropy to 32-bits, its impact over the WPA2 security can be observed in two instances – the first is the generation of the ANonce, while the second is the creation of the group master key (GMK).

The nonces’ purpose is to add liveness to the keys (we know that the PTK is derived from the PMK and the MAC addressed of the AP and the supplicant, but we also need something else to make sure that the temporal keys will change every time) to prevent replay attacks. The reduces entropy on the AP will make it easier (but not extremely easy!) to impersonate the AP through replay attacks.

In case of the generation of the GMK, things get trickier. The GMK is used to derive the GTK (using a nonce – which is going to also have a 32-bits entropy – and the MAC address of the AP), which is the encryption and integrity checking key. A 32-bit key is considered to be very weak nowadays, so the GTK will be fairly easy to brute force, making the multi- and broadcast communication easy to compromise.

(d)

Red Hat, despite being the most trusted vendor when discussing Linux and open-source technology, was also affected by the CVE-2017-13080 vulnerability which opened the door for the “key reinstallation attacks" (KRACKs)[1]. The impacted products were Red Hat Enterprise Linux 5, 6 (full list available here) and 7 (full list available here) – servers, desktops and other workstations being all exploitable.

Given the emergence of the wireless (data center) networks, as a solution to minimize time, efforts and deployment and maintenance costs for the wires and the severity of a successful KRACK attack, the issue was classified as “Important”, with an 8.1 base score and a high impact over confidentiality and integrity. Patches fixing the affected *wpa_supplicant* package were released for versions 6 and 7 in October 2017, but Red Hat Enterprise 5 was already in its extended life-cycle support phase, meaning that no bug or security fixes were provided – upgrading to a supported product version being the only option. Today, we can consider Red Hat Enterprise to be no longer (knowingly) vulnerable.

3. Mobile Consultant

(a)

Malicious adversaries can usually be classified into insiders and outsiders. The insiders originate from within the business (employees, contractors, business partners and compromised internal accounts), while outsider threats originate from external sources. While it's true that outsider attackers are more prevalent, the malicious insiders do generate higher revenue losses.

Usually, when we're talking about insiders, we think about physical theft, social engineering, privilege abuse, unintentional data leaks and copying sensitive data to personal accounts. Outsider attacks usually imply social engineering, malware, denial of service attacks, external forces generally needing more technical skills to do harm.

A few examples of actions an insider can take (and the attacks they can generate) are:

- misuse of audit tools (use the network monitoring techniques to extract sensitive information about users);

- abuse of privileges (gain access to network components to perform other types of attacks, for example use a base station and launch a DoS or signaling storm);

- manipulation of hardware equipment (include concealed software or hardware in an equipment during implementation or maintenance to install a backdoor for future use).

A few examples of attacks an outsider can generate are:

- phishing attacks (smishing) – trick the recipient of a message to click a link, send private information or download malware;

[1] https://access.redhat.com/security/vulnerabilities/kracks

- Denial of Service attacks – send a high number of authentication requests in a very short time until the network collapses and authorized devices experience loss of connectivity;

- channel jamming – transmit signals on the same radio frequencies as user equipment, disrupting the communication between phones and base stations;

- eavesdropping & message forgery – capture, analyze and tamper network communication information;

- LTE Evil Twin / fake access network node – masquerade as a legitimate base station to launch other types of attacks such as man-in-the-middle or network traffic manipulation;

- cellular frauds (port-out scams, SIM cloning, billing frauds).

(b)

The scheme can't be successful when the user equipment is in roaming because PrivTel Ro uses a proprietary framework. A fix would be to use a standardized scheme (such as EAP-AKA).

(c)

If the *rand* value generated by the home network is sent in plaintext and the user equipment uses its private key to sign the value *IMSI || rand*, then anyone can use the user equipment's public key (which is public) to gain access to the *IMSI || rand* value, detach the *rand* value and obtain the *IMSI*.

(d)

The IMSI protection scheme used by PrivTel Ro is simple and ineffective, based on (reversible) signatures, while in 5G, the concealment of the SUPI (the 5G networks equivalent of the IMSI) into SUCI is created in such a way (using public key cryptography) that only the home network can reverse the SUPI to SUCI transformation (using its private key).