More MitM Makes Mana Mostly Mediate Mischievous Messages
we

[hack, build, train, scan | ]

stuff

@singe
@cablethief
dominic@sensepost.com
michael@sensepost.com
Tracking Scenarios

Scenario 1 Snoopy
Don’t go to them
Make them come to you
same signature

wifi4:probe,1....

random MAC

actual MAC

SSID Probes
Enterprise EAP
Networks

Scenario 2 EAP
celestebarber Cutting edge
#celestechallengeaccepted #justinbieber #nailedit #funny @dailymail

Load more comments

kmcghehey @kitkatya123 tats
biankabajso @mtimeijja
ardita.sokoli @afoorfm
megancmcg @abrock11 I feel like you'll find this as entertaining as I do! The whole account meaning.
maxx.headroom @rebeccamorgan10
tefi_salazar "1+1=2" JAJAJAJAJA @juanisaldago @ipj_32
isajaramilloh JAJAJAJAJAJAJA @tefi_salazar
olaaatz21 @martin_el_dios_del_sexo
jaahraeez @luviv_juyo_brabo 1+1=2

8,315 likes
AUGUST 15, 2015
4-way handshake

association

Server Cert

Client Cert

4-way handshake
EAP Relay with Sycophant

By @cablethief
association

outer TLS setup

MSCHAPv2 challenge

MSCHAPv2 response

4-way handshake
4-way handshake

outer TLS setup

association

sycophant

MSCHAPv2 challenge

MSCHAPv2 response

4-way handshake

outer TLS setup

association
7.4. Man-in-the-Middle Attacks

Where EAP is tunneled within another protocol that omits peer authentication, there exists a potential vulnerability to a man-in-the-middle attack. For details, see [BINDING] and [MITM].

As noted in Section 2.1, EAP does not permit untunneled sequences of authentication methods. Were a sequence of EAP authentication methods to be permitted, the peer might not have proof that a single entity has acted as the authenticator for all EAP methods within the sequence. For example, an authenticator might terminate one EAP method, then forward the next method in the sequence to another party without the peer’s knowledge or consent. Similarly, the authenticator might not have proof that a single entity has acted as the peer for all EAP methods within the sequence.

Tunneling EAP within another protocol enables an attack by a rogue EAP authenticator tunneling EAP to a legitimate server. Where the tunneling protocol is used for key establishment but does not require peer authentication, an attacker convincing a legitimate peer to connect to it will be able to tunnel EAP packets to a legitimate server, successfully authenticating and obtaining the key. This allows the attacker to successfully establish itself as a man-in-the-middle, gaining access to the network, as well as the ability to decrypt data traffic between the legitimate peer and server.

This attack may be mitigated by the following measures:

[a] Requiring mutual authentication within EAP tunneling mechanisms.

[b] Requiring cryptographic binding between the EAP tunneling protocol and the tunneled EAP methods. Where cryptographic binding is supported, a mechanism is also needed to protect against downgrade attacks that would bypass it. For further details on cryptographic binding, see [BINDING].

[c] Limiting the EAP methods authorized for use without protection, based on peer and authenticator policy.

[d] Avoiding the use of tunnels when a single, strong method is available.
7.4. Man-in-the-Middle Attacks

Where EAP is tunneled within another protocol that omits peer authentication, there exists a potential vulnerability to a man-in-the-middle attack. For details, see [BINDING] and [MITM].

As noted in Section 2.1, EAP does not permit untunneled sequences of authentication methods. Were a sequence of EAP authentication methods to be permitted, the peer might not have proof that a single entity has acted as the authenticator for all EAP methods within the sequence. For example, an authenticator might terminate one EAP method, then forward the next method in the sequence to another party without the peer’s knowledge or consent. Similarly, the authenticator might not have proof that a single entity has acted as the peer for all EAP methods within the sequence.

[b] Requiring cryptographic binding between the EAP tunneling protocol and the tunneled EAP methods. Where cryptographic binding is supported, a mechanism is also needed to protect against downgrade attacks that would bypass it. For further details on cryptographic binding, see [BINDING].
Allow access only to those clients that authenticate with the specified methods.

EAP types are negotiated between NPS and the client in the order that they are listed.

EAP Types:
- Microsoft: Protected EAP (PEAP)

Less secure authentication methods:
- [ ] Microsoft Encrypted Authentication version 2 (MS-CHAP v2)
- [ ] User can change password after it has expired
- [ ] Microsoft Encrypted Authentication (MS-CHAP)
- [ ] User can change password after it has expired
- [ ] Encrypted authentication (CHAP)
- [ ] Unencrypted authentication (PAP, SPAP)
- [ ] Allow clients to connect without negotiating an authentication method

Edit Protected EAP Properties

Select the certificate the server should use to prove its identity to the client. A certificate that is configured for Protected EAP in Connection Request Policy will override this certificate.

Certificate issued to: WIN-JQGT06ARK9.wifidomain.local
Friendly name: WIN-JQGT06ARK9.wifidomain.local
Issuer: wifidomain-WIN-JQGT06ARK9-CA
Expiration date: 2019/06/30 5:06:48 PM

- [ ] Enable Fast Reconnect
- [ ] Disconnect Clients without Cryptobinding

Eap Types

Secured password (EAP-MSCHAP v2)
When connecting:

- Verify the server’s identity by validating the certificate

- Connect to these servers (examples: srv1; srv2; \srv3\com):

- Trusted Root Certification Authorities:
  - Entrust Root Certification Authority

- Notifications before connecting:
  - Tell user if the server’s identity can’t be verified

Select Authentication Method:

- Secured password (EAP-MSCHAP v2)
- Enable Fast Reconnect
- Disconnect if server does not present cryptobinding TLV
- Enable Identity Privacy
Mallory in the Middle

Scenario 3 MitM
Rail Gun

- Casing opens on mechanical arms and is released via button on rifle
- Stabilising fins
- Amour piercing
- High explosive
- Projectile casing (ejected when reloading)
Practise HW-less CTFs
https://w1f1.net/
@sensepost
@singe
@cablethief