Crypto Binding Revisited
draft-hartman-emu-mutual-crypto-binding

EMU
IETF 83
Sam Hartman
Dacheng Zhang
Margaret Wasserman
Tunneling the EAP Server

- Today, peers typically place little trust in the EAP server beyond protecting credentials.
- Channel Bindings, NEA and future extensions trust information returned from the server.
- Tunnels provide a way to integrate this into EAP.
Tunnel Security

- Clients often use certificates to identify tunnel servers

  Significant past focus on avoiding an attacker using a tunnel to capture the keys: tunnel MITM attack
Classic Tunnel Attack

1. Peer Initiates Connection to a Service
2. Tunnel Establishment
3. Non-Tunneled Method
4. Tunneled Authentication Method
5. MSK Keys
6. Attacker Connected as Peer
New Services and Tunnel Security

- Channel bindings extends the EAP threat model:
  - One NAS is not the same as another
  - We need the channel binding response from the right server
- Other new EAP services similarly involve the peer trusting the server
Server Insertion Attack

Peer Initiates Connection to a Service
(Intercepted by an Attacker)

Tunnel Establishment

Peer Connected to Attacker

Tunneled Method

Non-Tunneled Authentication Method
(Same as Inner Method from Tunnel)
MSK-based Keys

Service
But we fixed this, right?

- Crypto binding solves this, right?
  - Crypto binding may not confirm server to peer
  - Besides we just gave the attacker the MSK which we'll use for crypto binding
- Certificates solve this?
- Policy solves this?
Pop Quiz: EAP and Certificates

- Do all your EAP peers validate certificates back to a trust anchor?
- Do all your EAP peers know what subject name they expect in the certificate?
- Do your EAP peers rcheck to subject name?
- Yes to all questions is very rare
Challenges with EAP Certificate Validation

- Most EAP methods don't specify naming rules
- Certificate validation is only a SHOULD in many methods
- User interfaces make trust anchor configuration difficult
Policy Insufficient
Tunnel within Tunnel Attack

Peer Initiates Connection to a Service
(Intercepted by an Attacker)

Tunnel Establishment

Peer Connected to Attacker

Tunneled Method

Non-Tunneled Authentication Method
(Same as Inner Method from Tunnel)

MSK-based Keys
Tunnel to Tunnel Attack

Peer Initiates Connection to a Service
(Intercepted by an Attacker)

Peer Connected to Attacker
EMSK Crypto Binding

- The EMSK can be used to perform crypto binding
- Advantage: when it works provides transparent security with no additional config
- Only works with inner methods that support EMSK
- Not a complete solution
Server Certificate Received

- Cert Validated with CA?
  - Yes
    - Name Matches?
      - Yes: Server Authenticated
      - No: Continue...
  - No: Continue…
Inner Method Succeeds

- Server Auth’d by Inner Method
  - Yes
  - No
- Crypto Bind Success?
  - Yes
  - No
- EMSK Used?
  - Yes
  - No
- Tunnel Compromised Fail Immed.
- Continue...
- Server Authenticated
Server Authentication Required

- Server Auth’d
  - Yes
  - No
  - Policy Reqs Auth?
    - Yes: Report Failure
    - No: Continue...

- No
  - Yes: Continue...

Recommendations

- No one solution is sufficient
- Improve certificate handling
- Support EMSK crypto binding
- Find additional solutions
Feedback Desired

- Questions? Comments?
- Should we adopt draft-hartman-emu-mutual-crypto-binding to document this problem?