MEDIA ACCESS CONTROL (MAC) ADDRESS SPOOFING ATTACKS AGAINST PORT SECURITY

Andrew Buhr, Dale Lindskog, Pavol Zavarsky, Ron Ruhl
Concordia University College of Alberta
Findings

- Port Security is ineffective at preventing 3 different MAC Spoofing attacks in broadcast domains that span multiple switches.
- Port Security actually decrease the difficulty for 2 of these attacks.
Overview

- Background
  - Switch learning process
  - Port security
- Describe 2 attacks
  - Details, ease and limitations
- Discuss 3 countermeasures
  - Trunk port security
  - Port security sticky
  - Segregation mitigation strategy (recommended)
Not Covered in Presentation

- Third attack in a more sophisticated topology (Full MITM with three edge switches)
- Attack limitation details
  - Reconnaissance
  - Improving attack success
What is Cisco Port Security?

- Restrictive control applied to edge ports
- CAM overflow attacks -> MAC address spoofing
- Source MAC address compared to other learnt addresses
Non-secure Switch Learning Process

- Source MAC learning
- 1:N(int-MAC)
- Aging

![Diagram of MAC table and switch]
Secure Switch Learning Process

- Secure source MAC learning
- Non-aging
- Precedence
Interswitch Connections

**Directly Connected**

<table>
<thead>
<tr>
<th>MAC Table</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MAC</td>
<td>Port</td>
<td></td>
</tr>
<tr>
<td>a</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

**Indirectly Connected**

<table>
<thead>
<tr>
<th>MAC Table</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MAC</td>
<td>Port</td>
<td></td>
</tr>
<tr>
<td>a</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

**Diagram**

- DST(x) and SRC(a) represent the source and destination devices.
- Port(1) and Port(10) represent the connection ports.
- The MAC table shows the mapping of MAC addresses to ports.
MAC Spoofing
“The maximum number of secure MAC addresses have been added to the address table, and a station whose MAC address is not in the address table attempts to access the [secure] interface” - Cisco

- Mitigates CAM overflow attacks
- Caveats (in regards to MAC spoofing)
  - Legitimate MAC – no mechanism
  - Immediate registration – no mechanism
Port Security - Violation Condition (2)

- “An address learned or configured on one secure interface is seen on another secure interface in the same VLAN” - Cisco
- Mitigates MAC Spoofing
- Applies only when both interfaces are secure
Port Security Best Practices

- Enterprise Environment
- For a “dynamic environment, such as an access edge, where a port may have port security enabled with the maximum number [secure] MAC addresses set to one, enable only one [secure] MAC address to be dynamically learnt at any one time” – Cisco
Assumptions

(1) Attacker hasn’t registered MAC;
   - Or can unplug cable (clear secure MAC entry)
   - Sticky – more later

(2) No port security on interconnecting interfaces
   - Against best practices
   - More later

- We assume full network knowledge
  - Covered in limitations section
Attack #1 – Impersonation (initial)

- Port Security enabled on edge ports
- A listens for an ARP-Request \( V1 \rightarrow V2 \)
- \( V2 \) replies to \( V1 \)
- \( E1 \) MAC Address Table (initial):

<table>
<thead>
<tr>
<th>VLAN</th>
<th>MAC Addr</th>
<th>Type</th>
<th>Ports</th>
<th>Secure</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( V1 )</td>
<td>DYNAMIC</td>
<td>Fa0/1</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>( V2 )</td>
<td>DYNAMIC</td>
<td>Gi0/1</td>
<td>No</td>
</tr>
</tbody>
</table>
Attack #1 (resulting)

- A replays V2 exact ARP-Reply to update MAC address table
- No violation is thrown because initial V2 entry was non-secure and secure entries take precedence
- E1 MAC Address Table (resulting):

<table>
<thead>
<tr>
<th>VLAN</th>
<th>MAC Addr</th>
<th>Type</th>
<th>Ports</th>
<th>Secure</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>V1</td>
<td>DYNAMIC</td>
<td>Fa0/1</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>V2</td>
<td>DYNAMIC</td>
<td>Fa0/2</td>
<td>Yes</td>
</tr>
</tbody>
</table>

- All frames V1 -> A
- A cannot -> V2
Attack #1 (ease – no port security)

- Race condition introduced:
- If A replays V2 ARP-Reply, then E1 MAC Address Table will show V2 on Fa0/2
- But if V2 tries to communicate with any node on E1, then V2 will switch back to Gi0/1 on E1
- MAC table updates on last observed basis
- Port security locks in the MAC
Attack #1 (limitations)

- **A** cannot impersonate directly connected node - violation
- **A** cannot impersonate 2 indirectly connected nodes
- Can impersonate \( \frac{1}{2} \) network nodes and \( \frac{1}{4} \) of total communication streams

<table>
<thead>
<tr>
<th>A</th>
<th>V1</th>
<th>V2</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>E1</td>
<td>E1</td>
<td>E1</td>
<td>Port security violation</td>
</tr>
<tr>
<td>E1</td>
<td>E1</td>
<td>E2</td>
<td>Impersonate V2 (V1 perspective)</td>
</tr>
<tr>
<td>E1</td>
<td>E2</td>
<td>E1</td>
<td>Impersonate V1 (V2 perspective)</td>
</tr>
<tr>
<td>E1</td>
<td>E2</td>
<td>E2</td>
<td>No port security violation</td>
</tr>
</tbody>
</table>
Attack #2 – Full MITM

- Additional switch access
- \textbf{A} replays ARP-Reply out Fa0/2 on \textbf{E1} to poison \textbf{E1} (same as Attack #1)
- \textbf{A} then replays ARP-Request out Fa0/2 on \textbf{E2} to poison \textbf{E2}
- Removes limitation of spoofing directly connected nodes (attack victims doubled)
May be detected because ARP-Reply is unsolicited (could be blocked)

Attack is more difficult without port security because race conditions exit on both sides

½ of communication streams (no direct to direct)
Defences and Countermeasures (1)

(1) Interconnecting Switch Port Security
- Would span secure entries across broadcast domain
- Etherchannel is not supported
- STP is not interoperable
  - Topology change – different ports
- Node relocation problems
  - No deregistration mechanism (distribution lock)
- Increased risk to infrastructure
Defences and Countermeasures (2)

(2) Port Security Sticky

- More difficult to spoof if address already registered
- Node relocation problems
  - Deliver to wrong port
  - Manual change process control
- Undermines dynamic benefit of switch learning process
Defences and Countermeasures

(3) Segregate broadcast domains based on trust and role

- Ideal to de-span all broadcast domains
  - Prevents attacks
- But logical grouping is sometimes required
  - Flexibility
  - Cost
  - Performance
Defences and Countermeasures (3)

- Segregate trusted from untrusted
  - Then they can’t attack each other
Defences and Countermeasures (3)

- Segregate untrusted nodes from untrusted nodes
  - They are the most likely to attack

- Segregate trusted based on role (client or server)
  - Trusted clients can still span
  - Trusted servers can either span or not
    - Implement sticky when they span