

Network Admission Control Segen oder Fluch?

Dror-John Roecher



Ihr Referent: Dror-John Röcher

- **Jahrgang 1973**
- **1996 – 2004: freiberuflich tätig als Referent & Berater mit Schwerpunkten “Enterprise Networking” & “Network Security”**
- **Seit 2004: Senior Security Consultant & Gesellschafter, ERNW GmbH**
- **Ausgewählte Vorträge/Publicationen:**
 - Security Patchmanagement, IIR Security Forum, 2005
 - Blackberry Security & Mobile Security, itsecurity, 2006
 - OSPF Security: Vortrag & Tool-Release, IT-Underground, Prag, 2007
 - NAC@ACK – Hacking the Cisco NAC Framework, Vortrag & Tool-Release, Blackhat 2007, Amsterdam
- **Kontakt Daten:**
 - Email: droecher@ernw.de
 - Mobil: 0173-6745905



Agenda

- **Part 1 – Introduction (very short)**
 - Some marketing buzz on NAC
- **Part 2 – NAC Technology**
 - All you need to know about NAC (in order to hack it)
- **Part 3 – Security Analysis**
 - Delving into the security flaws of Ciscos' NAC solution
- **Part 4 – Approaching NAC@ACK**
 - The stony road towards a working exploit
- **Part 5 – Some thoughts on mitigation**



Cisco NAC ist "gehackt" ,-) – Demo am Ende des Vortrags



heise Security - News - Cisco Netzwerkzugangskontrolle NAC ausgetrickst - Mozilla Firefox

Datei Bearbeiten Ansicht Chronik Lesezeichen Extras Hilfe

http://www.heise.de/security/news/meldung/87663

heise online · c't · iX · Technology Review · Telepolis · mobil · **Security** · Netze · heise open · heise resale · c't-TV · Jobs · Kiosk

heise Security

Sie sind Gast
Einloggen | Registrieren

Suche

los

News

- 7-Tage-Alerts
- 7-Tage-News
- News-Archiv
- Newsletter
- English News

News

Meldung vom 30.03.2007 15:42 [<< Vorige] [Nächste >>]

Cisco Netzwerkzugangskontrolle NAC ausgetrickst

Sicherheitsexperten haben auf der [Black-Hat-Konferenz](#) in Amsterdam gezeigt, wie sich [Ciscos](#) Netzwerkzugangskontrolle [NAC](#) austricksen lässt. Mit einem modifizierten Trust Agent war es Michael Thumann und Dror-John Röcher von [ERNW](#) in einer Live-Demonstration möglich, sich mit einem nicht den Netzwerk-Policies genügenden Rechner vollen Zugang zu einem NAC-gesicherten Netzwerk zu verschaffen.

Cisco hat das Problem laut Thumann und Röcher bereits bestätigt und will in Kürze einen eigenen Fehlerbericht dazu herausgeben. Mit Systemen wie Cisco NAC können Netzwerkadministratoren

Scripts Partially Allowed [<script>: 5] [J+F+P: 0]

Fertig 193.99.144.85 Scripts Partially Allowed [<script>: 5] [J+F+P: 0] McAfee SiteAdvisor



Part 1 - Introduction



NAC Goal: To Control Clients

- **Current Patch-Level**
- **Up-to-date AV-Protection**
- **No unwanted programs (e.g. I33t-t001\$)**
- **Measure & Enforce Compliance to Security Policy**
- **Raise Security Level**
- **(Increase user-awareness ?)**



Where could it be used?

- **Wherever clients connect to the network:**
 - RAS (VPN, Dial-in)
 - SSL-VPN
 - Within the wired H-Net at all locations
 - On wireless segments, too



Some Major Players

- **Cisco:** The Cisco NAC is at the core of the „self defending network“. Cisco NAC can be considered a framework and has broad third-party support (more than 80 vendors support it).
- **Microsoft:** Microsoft has developed an approach named „Network Access Protection“ (NAP) which is supposed to ship with Vista and Longhorn.
- **Checkpoint:** Has a working solution which is build around its „Integrity client“ [think: SecuRemote].



General Concept

- **The general concept is common to most vendor-solutions:**
 1. Detect new clients
 2. Determine their state
 3. Compare „state of client“ to a set of criteria and derive „access level“
 4. Enforce access level
 5. Monitor „state of client“ and react to changes

- **Some vendors have a different approach (namely Consentry) which will be covered later.**



- **State can be determined from:**
 - Operating System Version
 - OS Patch Level
 - Existence of a „Network Access Agent“
 - Up-to-date AV-Software
 - Desktop-Firewall state & policy
 - Version & Patchlevel of Applications (think IE)
 - Absence of unwanted programs (e.g. I337-7001)
 - ...



Limit network access based on state of client.

■ Limitation can be:

- User-ACL
- Dynamic VLAN assignment
- URL Redirection
- Quarantine Network / Remediation Access only
- No Access
- Full Access



Enterprise Operational Requirements

- **Central Management**
- **Definition of Whitelists**
- **Integration into existing network infrastructure.**
- **Integration into existing procedures/processes.**
- **Agentless support.**
- **Agent available on many OS.**
- **Preferrably „open standard“ and broad third-party support.**
- **Compatibility with one of the major frameworks (cisco NAC, TCP-TNC)**
- **Scalability to Enterprise-Size**
- **Auto-Remediation**



General reasons in favour of NAC

- **Can raise security level.**
- **Could possibly be required by „compliance“ department.**
- **Could leverage Enterprise-wide 802.1x deployment.**
- **Could become a „must have“ like SSL-VPN (management pressure, Gartner „magic quadrant)**
- **Could lead to increased awareness.**



General Reasons against NAC

- **Market still developing. Landscape may look completely different in 3-5 years.**
- **Added complexity in network.**
- **Costs.**
- **Infrastructure not ready (think 802.1x)**
- **Support for mobile devices beyond Laptops not yet available (Windows Mobile 5|6, Blackberry, etc)**



In Focus: Cisco NAC

Most comprehensive & best-known solution on the market



Why is Cisco selling Cisco NAC?

- Because customers are willing to pay for it , -)
- But why are customers willing to pay for it?
- Because Cisco makes some pretty cool promises... see next slide



From: <http://www.cisco.com/go/nac>

NAC Business Benefits

Dramatically improves security

- Ensures endpoints (laptops, PCs, PDAs, servers, etc.) conform to security policy
- Proactively protects against worms, viruses, spyware, and malware; focuses operations on prevention, not reaction

Extends existing investment

- Enables broad integration with multivendor security and management software
- Enhances investment in network infrastructure and vendor software
- Combining with Cisco Security Agent enables "trusted QoS" capabilities that classify mission-critical traffic at the endpoint and prioritize it in the network

Increases enterprise resilience

- Comprehensive admission control across all access methods
- Prevents non-compliant and rogue endpoints from impacting network
- Reduces OpEx related to identifying and repairing non-compliant, rogue, and infected systems

Comprehensive span of control

- Assesses all endpoints across all access methods, including LAN, wireless connectivity, remote access, and WAN

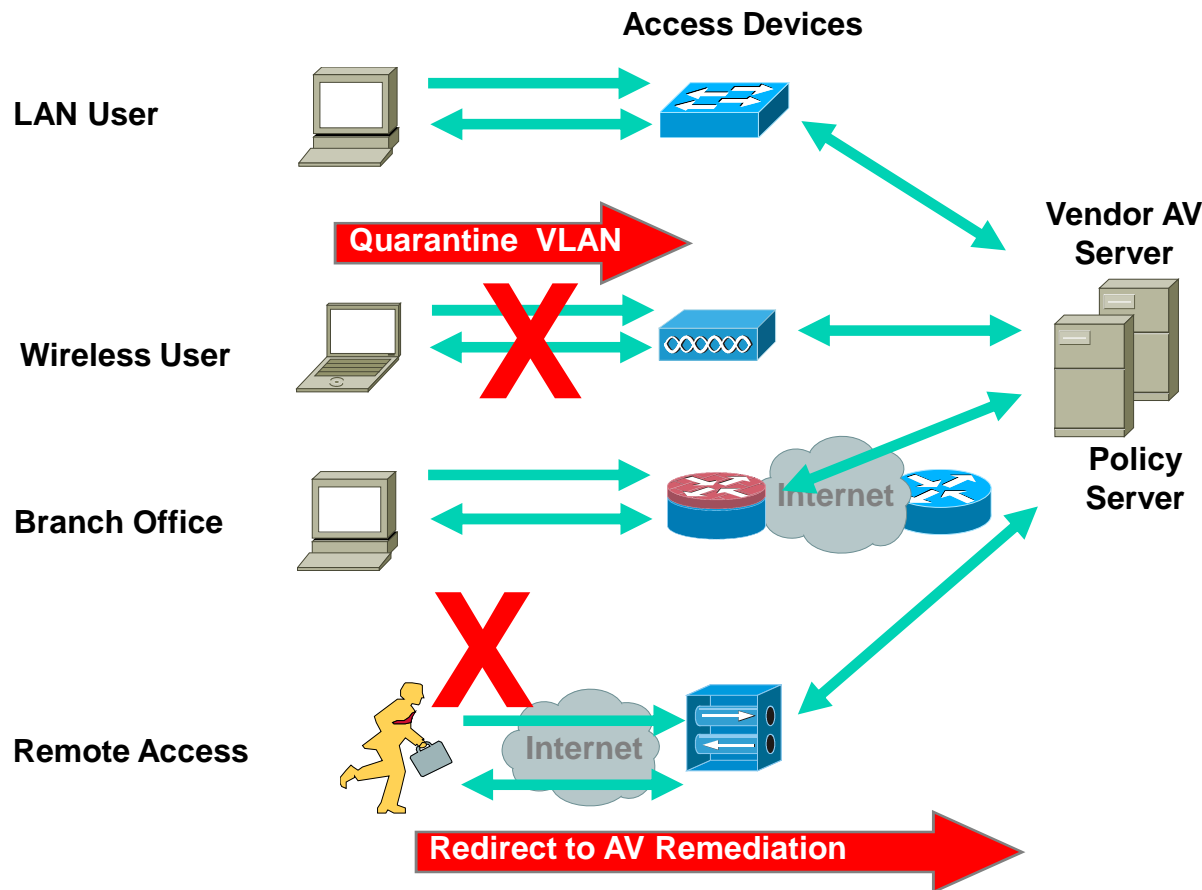


The idea behind Cisco NAC

- **Grant access to the network based on the grade of compliance to a defined (security) policy. So it is first of all a compliance solution and not a security solution.**
- **Security Policy can usually be broken down to:**
 - Patch level (OS & Application)
 - AV signatures & scan engine up to date
 - No „unwanted“ programs (e.g. I33t t00ls)
 - Desktop Firewall up & running
- **If a client is non-compliant to the policy [and is not whitelisted somewhere – think network-printers], restrict access.**



Policy based Access...



1. Access Device detects new client.
2. Access Device queries the client for an agent and relays information to a backend policy server.
3. Policy Server checks received information against defined rules and derives an appropriate access-level
4. Access-Device enforces restrictions

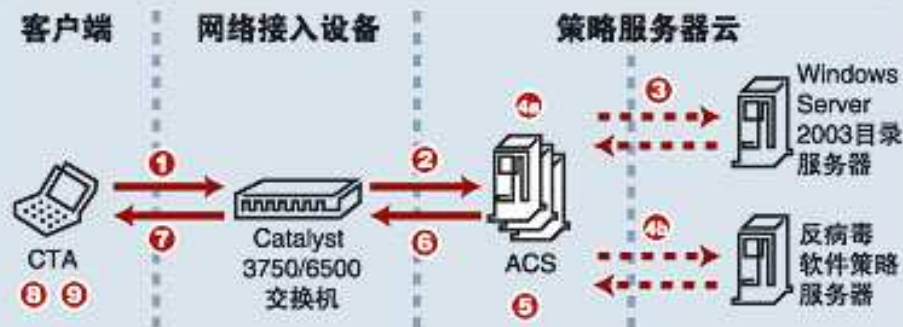


Part 2 – NAC Technology



What is Cisco NAC?

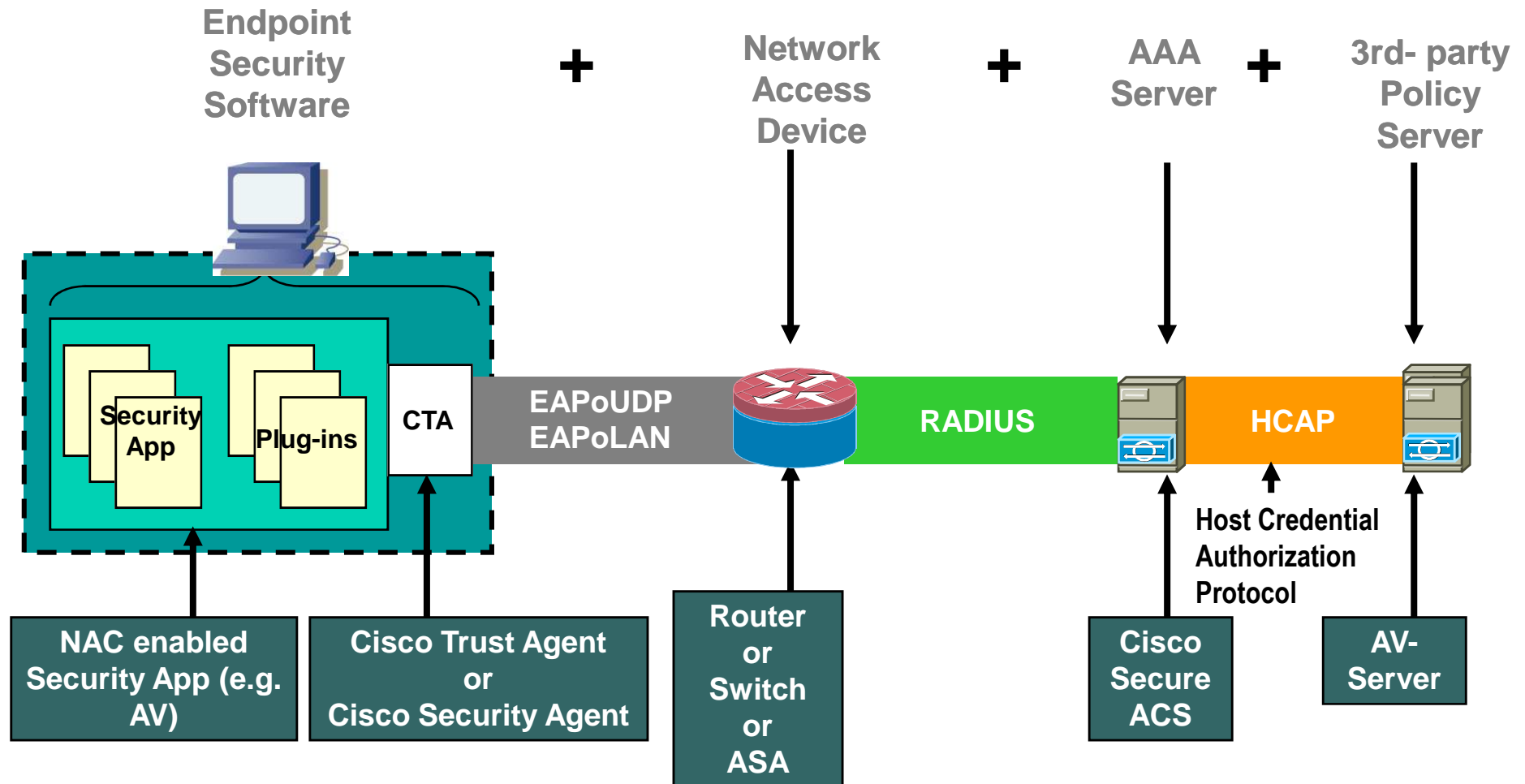
NAC over 802.1x工作原理



- ① CTA将身份认证信息和主机安全信息发给交换机（借助802.1x）。
- ② 交换机将认证信息发送给ACS。
- ③ ACS收到信息开始验证工作。与目录服务器交互，确认用户权限。
- ④ ACS检查入网计算机Service Pack, Hotfix, CSA版本等。
- ⑤ ACS与第三方反病毒策略服务器进行交互，确认用户的健康状况。
- ⑥ 根据AD和反病毒策略服务器反馈的信息进行判断，认证。
- ⑦ 根据验证的结果向交换机下发策略，若为健康计算机划分到VLAN 100，不健康计算机划分到隔离VLAN。添加每用户ACL。
- ⑧ 将认证结果告知终端上的CTA软件。
- ⑨ CTA获知计算机的状态，健康或不健康，是否通过认证。
- ⑩ CSA从CTA处获知计算机状态，并决定是否限制应用，并记录到系统日志，发送给MARS。



A „big overview“ picture...



There are 3 different NAC flavours...

- **NAC-Layer3-IP**
 - Access-restrictions are implemented as IP-ACLs
 - NAD is a Layer-3 device (e.g. a Router or a VPN-Concentrator/Firewall).
 - The communication takes place using PEAP over EAP over UDP (EoU).
- **NAC-Layer2-IP**
 - Access-restrictions as IP-ACLs on a VLAN-interface of a switch.
 - The communication takes place using PEAP over EAP over UDP (EoU)
- **NAC-Layer2-802.1x**
 - Uses 802.1x port control to restrict network access
 - Obviously the device enforcing these restrictions is a switch.
 - EAP-FAST is used in conjunction with 802.1x.
 - This is the only NAC flavour where the client is:
 - authenticated before being allowed on the network
 - restricted from communicating with its local subnet



(Some) Features...

Feature	NAC-L2-802.1x	NAC-L2-IP	NAC-L3-IP
Trigger	Data Link / Switchport	DHCP / ARP	Routed Packet
Machine ID	Yes	No	No
User ID	Yes	No	No
Posture	Yes	Yes	Yes
VLAN Assignment	Yes	No	No
URL Redirection	No	Yes	Yes
Downloadable ACLs	Cat65k only	Yes	Yes

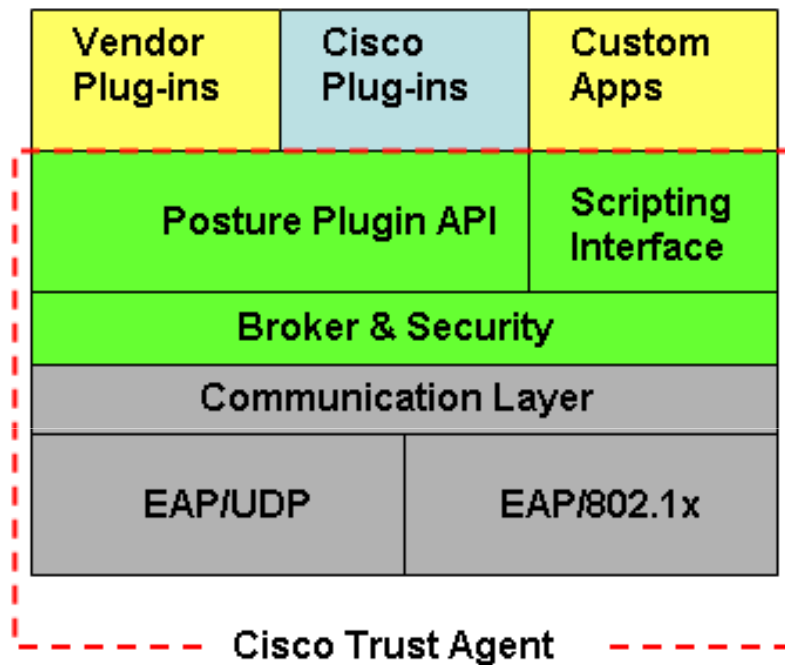


Yet another agent: Cisco Trust Agent

- **The Cisco Trust Agent (CTA) is the main component of the NAC framework installed on the clients.**
- **Its' tasks are to collect „posture data“ about the client and forward it to the ACS via the NAD.**
- **It has a plug-in interface for 3rd party vendors' NAC-enabled applications.**
- **It has a scripting interface for self-written scripts.**



CTA architecture



- **The CTA comes with two plug-ins by default:**
 - Cisco:PA
 - Cisco:Host



- **The information collected are Attribute-Value-pairs categorized by**
 - Vendor: ID based on IANA SMI assignement
 - Application-Type: see next slide
 - Credential Name: e.g. “OS Version”
 - Value-Format: String, Date, etc.
- **For all plug-ins & scripts this information is collected in a plaintext “.inf-file”.**



Application Types in Cisco NAC

Application-Type ID	Application-Type Name	Usage
1	PA	Posture Agent
2	Host / OS	Host information
3	AV	Anti Virus
4	FW	Firewall
5	HIPS	Host IPS
6	Audit	Audit
32768 – 65536		Reserved for “local use” (custom plug-ins or scripts)



Credentials for Cisco:PA & Cisco:Hosts

Application-Type	Attribute Number	Attribute Name	Value-Type
Posture Agent	3	Agent-Name (PA-Name)	String
	4	Agent-Version	Version
	5	OS-Type	String
	6	OS-Version	Version
	7	User-Notification	String
	8	OS-Kernel	String
	9	OS-Kernel-Version	Version
Host	11	Machine-Posture-State	1 – Booting, 2 – Running, 3 – Logged in.
	6	Service Packs	String
	7	Hot Fixes	String
	8	Host-FQDN	String



Posture Tokens...

- For each plug-in/Application/script an “Application Posture Token” (APT) is derived by the ACS through the configured policy.
- This token is one out of:
 - Healthy, Checkup, Quarantine, Transition, Infected, Unknown (see next slide for definitions of these tokens)
- From all APTs a “System Posture Token” (SPT) is derived – this corresponds to the APT which will grant the least access on the network to the client.
- The SPT is associated with access-restrictions on the ACS (e.g. downloadable ACL, URL-Redirection).



Posture Tokens – well defined

- **“Healthy”**: fully compliant with the admission policy for the specified application.
- **“Checkup”**: partial but sufficient compliance with the admission policy, no need to restrict access, a warning to the user may be issued.
- **“Transition”**: either during boot-time, when not all necessary services have been started or during an audit-process for clientless hosts, temporary access-restrictions may be applied.
- **“Quarantine”**: insufficient compliance with the admission policy, network access is usually restricted to a quarantine/remediation segment.
- **“Infected”**: active infection detected, usually most restrictive network access even up to complete isolation.
- **“Unknown”**: a token can not be determined or no CTA installed on client. This may lead to partial access (guest-vlan & internet-access for example).



Sample inf-File for Trendmicro AV

[main]

dll=tmabpp.dll
PluginName=tmabpp.dll
VendorID=6101
VendorIDName=TrendMicro, Inc
AppList=av

The name of the plug-in. In case of a script this would be ctascriptPP.dll and the vendor-id would be "Cisco" for scripts.

[av]

AppType=3
AppTypeName=Antivirus
AttributeList=attr1,attr2,attr3,attr4,attr5,attr6,attr7,attr8,attr9,attr10,attr11,attr12,attr13,attr14
attr1= 1, Unsigned32, Application-Posture-Token
attr2=2, Unsigned32, System-Posture-Token
attr3=3, String, Software-Name
attr4=4, Unsigned32, Software-ID
attr5=5, Version, Software-Version
attr6=6, Version, Scan-Engine-Version
attr7=7, Version, Dat-Version
attr8=8, Time, Dat-Date
attr9=9, Unsigned32, Protection-Enabled
attr10=10, String, Action

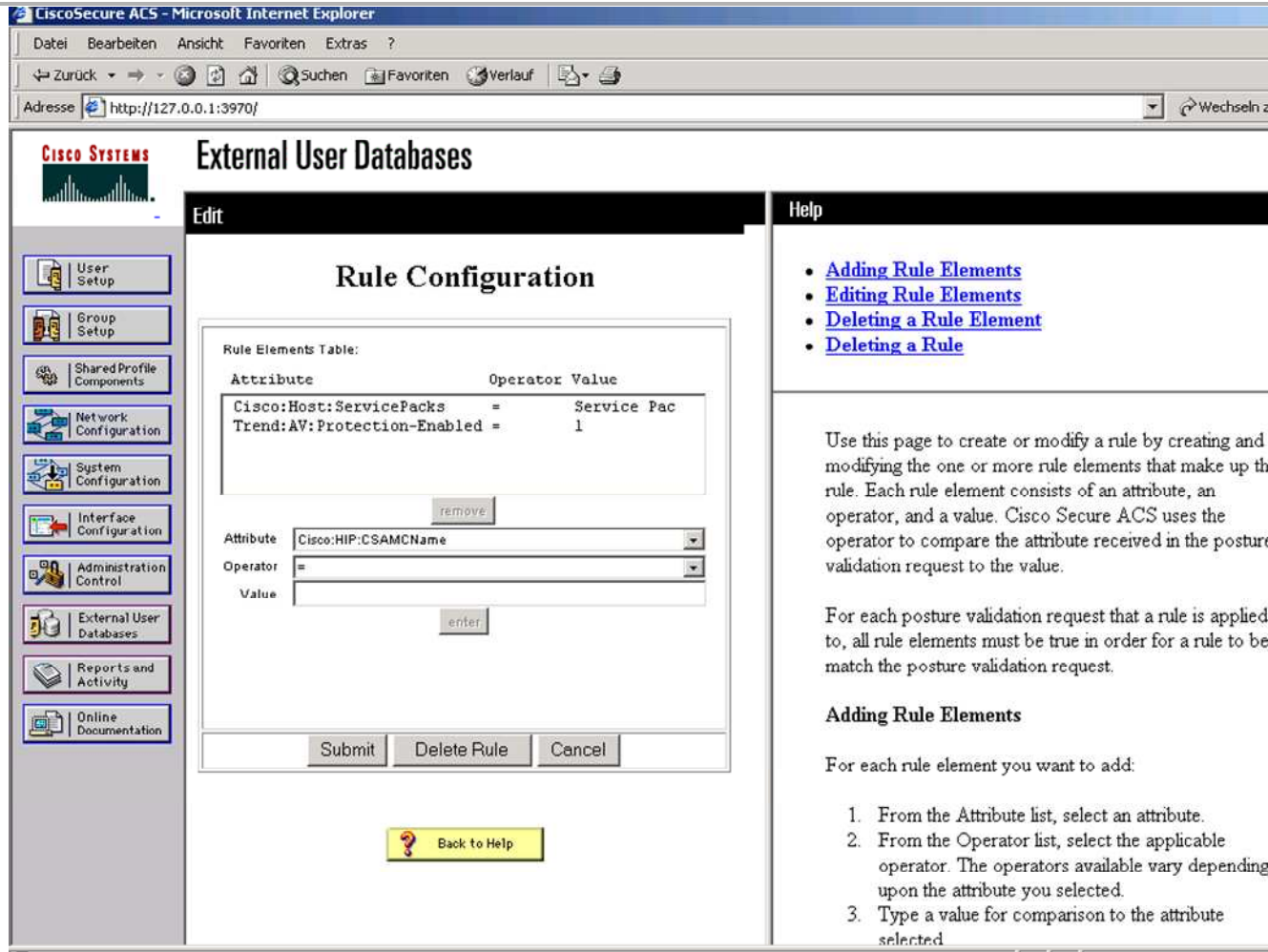
Official Credentials

attr11=32768, String, OSCE-Srv-Hostname
attr12=32769, OctetArray, Client-GUID
attr13=32770, Ipv4Address, Client-IP
attr14=32771, OctetArray, Client-MACddd

Private Credentials from the Vendor



Sample Policy on Cisco ACS



The screenshot shows the Cisco ACS web interface for configuring rules. The main content area is titled "Rule Configuration" and contains a "Rule Elements Table" with the following entries:

Attribute	Operator	Value
Cisco:Host:ServicePacks	=	Service Pac
Trend:AV:Protection-Enabled	=	1

Below the table, there is a form to add a new rule element with the following fields:

- Attribute: Cisco:HIP:CSAMCName
- Operator: =
- Value: (empty)

Buttons for "remove", "enter", "Submit", "Delete Rule", and "Cancel" are visible. A "Back to Help" button is located at the bottom of the configuration area.

The right-hand side of the page contains a "Help" section with the following links:

- [Adding Rule Elements](#)
- [Editing Rule Elements](#)
- [Deleting a Rule Element](#)
- [Deleting a Rule](#)

The help text explains that this page is used to create or modify rules by adding or removing rule elements. Each rule element consists of an attribute, an operator, and a value. It also provides instructions on how to add rule elements:

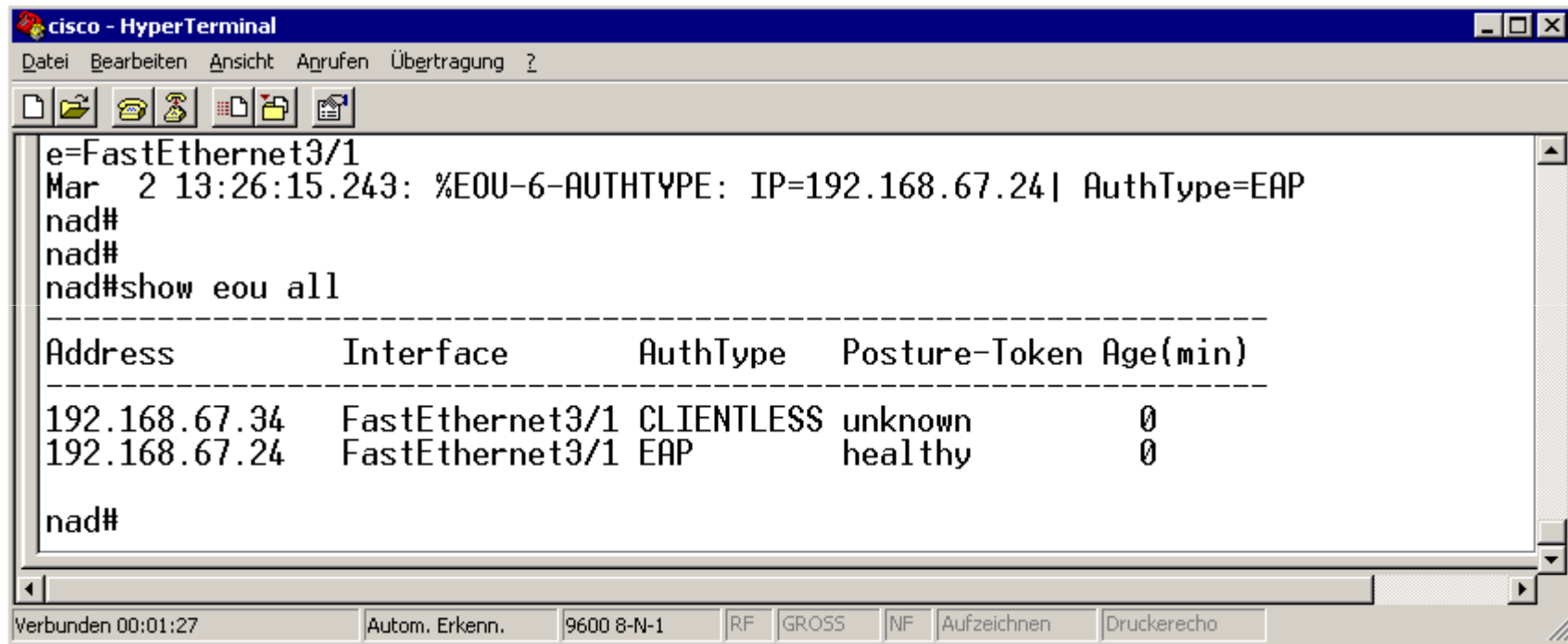
Adding Rule Elements

For each rule element you want to add:

1. From the Attribute list, select an attribute.
2. From the Operator list, select the applicable operator. The operators available vary depending upon the attribute you selected.
3. Type a value for comparison to the attribute selected.



And the resulting SPT on a NAD

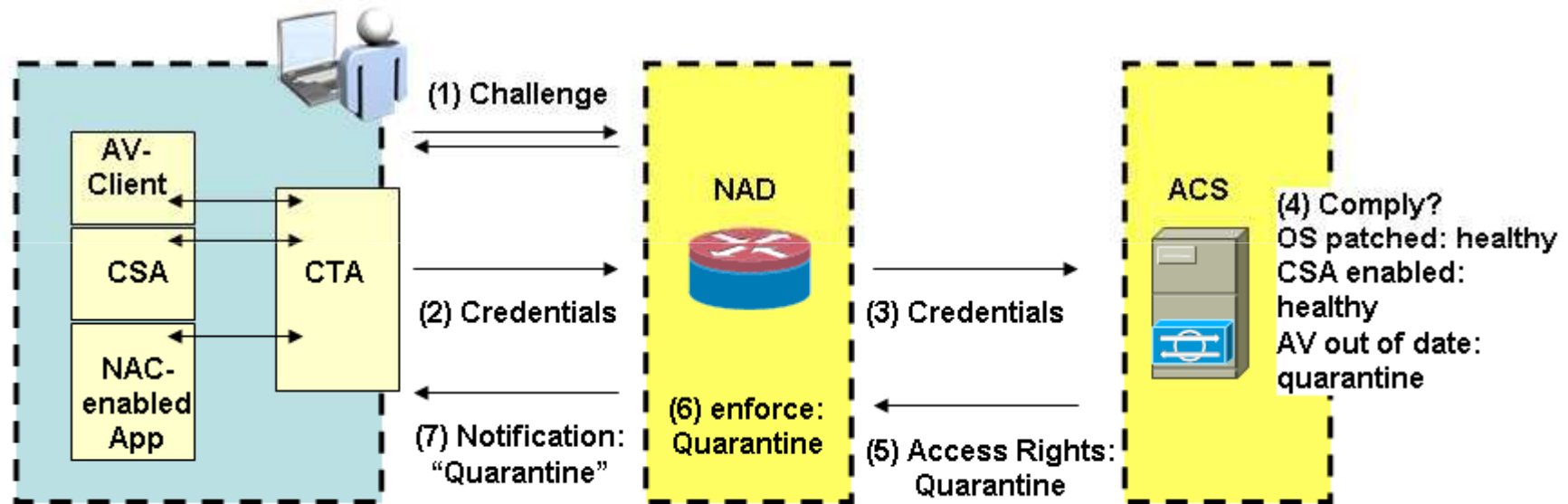


```
cisco - HyperTerminal
Datei Bearbeiten Ansicht Anrufen Übertragung ?
e=FastEthernet3/1
Mar  2 13:26:15.243: %E0U-6-AUTHTYPE: IP=192.168.67.24| AuthType=EAP
nad#
nad#
nad#show eou all
-----
Address          Interface      AuthType      Posture-Token Age(min)
-----
192.168.67.34    FastEthernet3/1 CLIENTLESS    unknown        0
192.168.67.24    FastEthernet3/1 EAP           healthy        0
nad#
```

Verbunden 00:01:27 Autom. Erkenn. 9600 8-N-1 RF GROSS NF Aufzeichnen Druckerecho



General Communication Flow



Transport Mechanisms...

- **NAC-Layer2-802.1x**

- Uses 802.1x
- Uses EAP-FAST as EAP method
- Uses EAP-TLV to transport posture information

- **NAC-Layer2-IP**

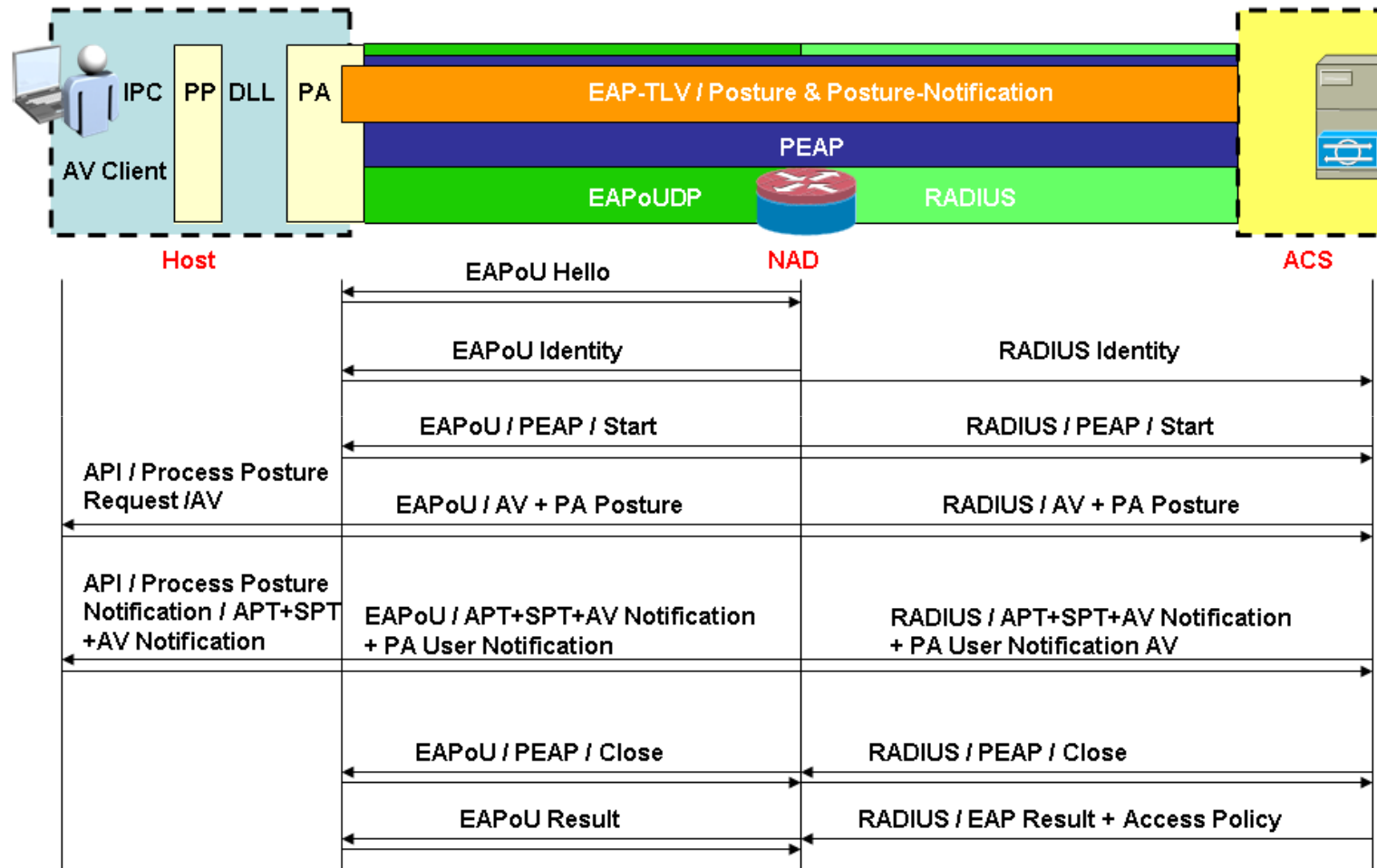
- Uses EAP over UDP (Port 21862 on client & NAD)
- Uses PEAPv1 as EAP method without inner authentication
- Uses EAP-TLV to transport posture information

- **NAC-Layer3-IP**

- Uses EAP over UDP (Port 21862 on client & NAD)
- Uses PEAPv1 as EAP method without inner authentication
- Uses EAP-TLV to transport posture information



NAC-L3-IP Communication Flow



Part 3 – Security Analysis



Flawed by Design 1: Client Authentication

	NAC-Layer 3 IP	NAC Layer 2 IP	NAC Layer 2 802.1x
Client Authentication	No intrinsic Client Authentication. In VPN scenarios there is a “VPN Authentication” which might be considered a “mitigating control”.	No intrinsic Client Authentication – and no means of “adding” such on top.	Client Authentication based on 802.1x/EAP-FAST
Restriction of access on local subnet.	It is not possible to restrict access to the local subnet via NAC.	It is not possible to restrict access to the local subnet via NAC.	Access to local subnet can be denied through “port shutdown” via NAC.



- **Second design flaw is somewhat related to the first flaw:**

Authorization without Authentication

- **This is clearly breaking a “secure by design” approach [for a security product] and is not conforming to “Best Current Practices”**



Flawed by Design Conclusion: Epimenides Paradox

- **Epimenides was a Cretan (philosopher) who made one statement: "All Cretans are liars."**
- **Same paradox applies to Cisco NAC as well:**
 - The goal is to judge the “compliance”-level of (un)known & untrusted clients.
 - This is achieved by asking the (un)known & untrusted client about itself.
 - How can the ACS be sure that the client is a Cretan philosopher (a liar)?



So what? Where is the attack?

Posture Spoofing Attack

- We define “posture spoofing” as an attack where a legitimate or illegitimate client spoofs “NAC posture credentials” in order to get unrestricted network access.



Attackers Definition - Insider

- **Insider:** An insider is a legitimate user of a NAC-protected network. The client has a working installation of the CTA and valid user/machine-credentials for the network. Additionally the inside attacker has the certificate of the ACS installed in its certificate store and if 802.1x is being used, this attacker has valid EAP-FAST-Credentials (PAC).
- The insider simply wants to bypass restrictions placed on his machine (e.g. no “leet tools” allowed and NAC checks list of installed programs).



Attackers Definition - Outsider

- **Outsider:** An outsider is not a legitimate user of the NAC-protected network and wants to get unrestricted access to the network. The outsider has no valid user/machine-credentials and no working CTA installation.



Attack Vectors

- **Code an “alternative” NAC client**
 - Definitely possible
 - Will not work on 802.1x with EAP-FAST for outsider.
 - Currently “development in process” 😊
- **Replace plug-ins with self-written ones**
 - Definitely possible (be patient for ~50 more slides *just kidding*)
 - Works for the “insider” but not for the “outsider”.
 - Less work than the “alternative client
- **Abuse the scripting interface**
 - Not verified yet – limitations on “Vendor-ID” and “Application-ID” apply and not (yet) known if these are enforced or can be circumvented
 - If possible – the easiest way 😊



Feasible Attack Vectors

	Insider	Outsider
NAC-L2-802.1x	DLL/Plug-In replacement Scripting Interface CTA replacement	None as to our current knowledge.
NAC-L2-IP	DLL/Plug-In replacement Scripting Interface CTA replacement	CTA replacement
NACL-L3-IP	DLL/Plug-In replacement Scripting Interface CTA replacement	CTA replacement



Part 4 – Approaching NAC@AK



The ugly stuff – working with a structured approach *sigh

- **Step 1: Define what you need to know in order to get it working.**
- **Step 2: Sketch an attack-tree showing steps towards the goal.**
- **Step 3: Evaluate the components of the attack-tree for feasibility. Get the “tools” & know the “techniques” you need.**
- **Step 4: Pursue the feasible steps from step 3.**
- **Step 5: loop to step (1) until you get it working ,-)**



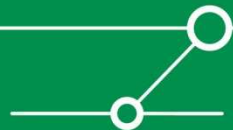
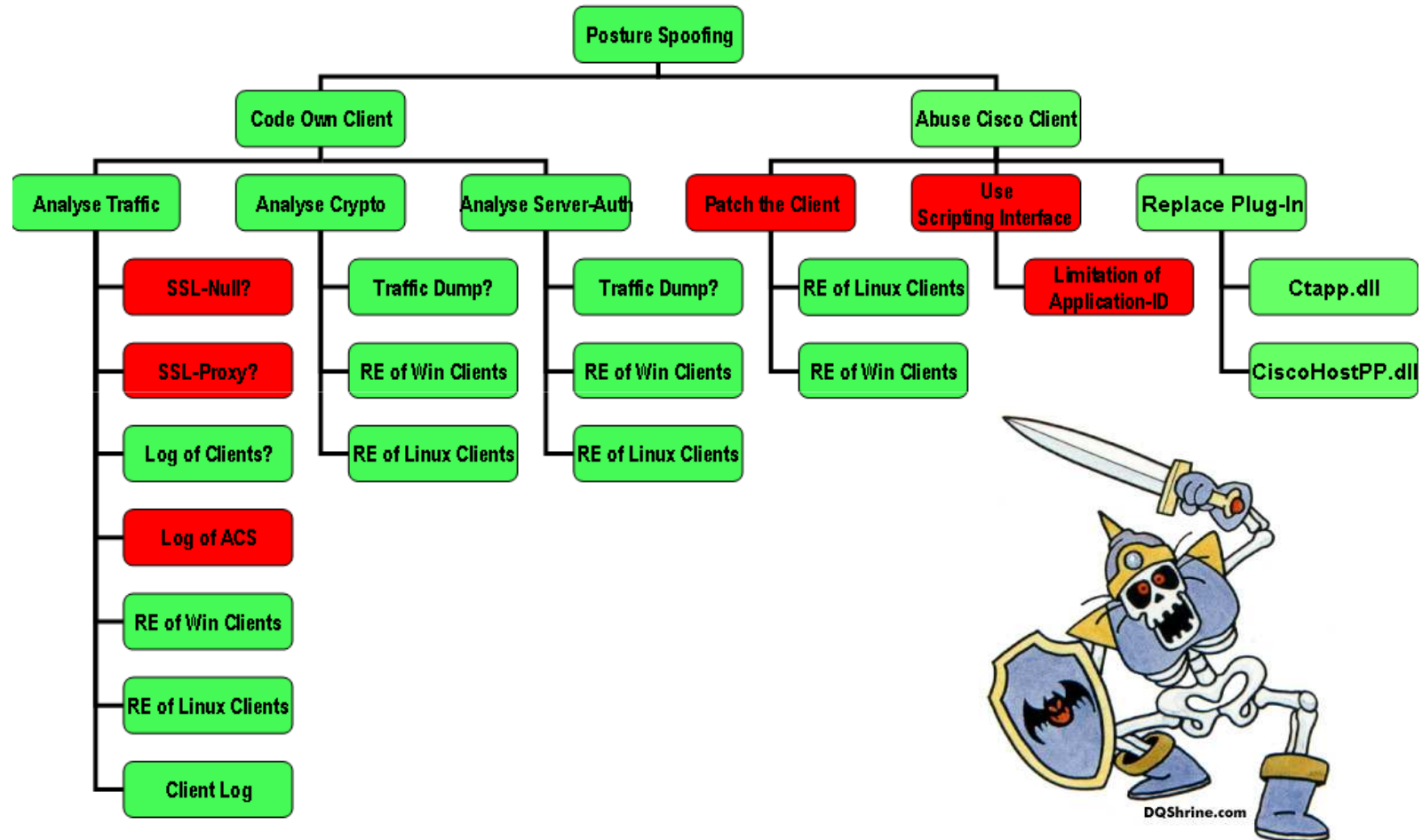
Want to know

■ Everything relating to...

- Communication flow
- Packet format
- Data-structures
- Used Crypto
- Used libraries
- Existing interfaces
- Program flow
- Used Authentication
- ...



Attack Tree



Tools & Techniques

- **Reverse Engineering**
 - Reverse Engineering aims at uncovering the constructional elements of a product. IDAPro ☺
- **Packet Sniffing**
 - You all know that - Wireshark/Ethereal
- **Packet Diffing**
 - Extracting common and differing parts of two packets.
- **Debugging / API-Monitoring / Function-Hooking**
 - Through attaching a debugger or api-monitor to the running process, it is possible to actually see the contents of the stack while the program is running.
- **Built-in capabilities**
 - Logging / Debugging capabilities of the product – Cisco is usually *_very_* good at that!
- **RTFM**
 - Read Read Read – often then vendor will tell you a lot about the product.



Big “want to have”: Cleartext Packets...

- **Communication is encrypted using TLS... packet capture shows encrypted packets.**
- **Not possible to get cleartext dump with tools (SSLProxy, etc.) – TLS over UDP not supported by tools.**
- **RTFM: Client Log can be enabled and it can dump cleartext payload of packets *g**



Packet Sniffing & Diffing

healthy04-hdb-labsetup.cap - Wireshark

Filter: Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
5	0.035843	192.168.81.66	192.168.94.100	UDP	Source port: 21862 Destination port: 21862
6	0.724066	192.168.94.100	192.168.81.66	UDP	Source port: 21862 Destination port: 21862
7	0.741727	192.168.81.66	192.168.94.100	UDP	Source port: 21862 Destination port: 21862
8	0.743379	192.168.94.100	192.168.81.66	UDP	Source port: 21862 Destination port: 21862
9	0.758508	192.168.81.66	192.168.94.100	UDP	Source port: 21862 Destination port: 21862
10	0.762045	192.168.94.100	192.168.81.66	UDP	Source port: 21862 Destination port: 21862
11	0.774664	192.168.81.66	192.168.94.100	UDP	Source port: 21862 Destination port: 21862
12	0.802075	192.168.94.100	192.168.81.66	UDP	Source port: 21862 Destination port: 21862
13	0.841586	192.168.81.66	192.168.94.100	UDP	Source port: 21862 Destination port: 21862
14	0.843672	192.168.94.100	192.168.81.66	UDP	Source port: 21862 Destination port: 21862

Frame 9 (1066 bytes on wire (1066 bytes captured))

- Ethernet II, Src: Cisco_9b:f7:c8 (00:14:f2:9b:f7:c8), Dst: Aironet_a7:48:5d (00:40:96:a7:48:5d)
- Internet Protocol, Src: 192.168.81.66 (192.168.81.66), Dst: 192.168.94.100 (192.168.94.100)
- User Datagram Protocol, Src Port: 21862 (21862), Dst Port: 21862 (21862)
- Data (1024 bytes)

```

0000 00 40 96 a7 48 5d 00 14 f2 9b f7 c8 08 00 45 00  .@..H].. ..E.
0010 04 1c 20 4a 00 00 fe 11 67 8f c0 a8 51 42 c0 a8  ..j...g..QB..
0020 5e 64 55 66 55 66 04 08 bf 02 00 13 03 f4 09 14  AduFuF.. ..
0030 eb 2e 64 54 c6 6d 80 02 03 f0 01 30 03 f0 19 41  ..dT.m.. ..0...A
0040 69 6c 6c 61 2e 6c 6f 63 61 6c 5c 43 65 72 74 45  illa.loc al\CertE
0050 6e 72 6f 6c 6c 5c 45 52 4e 57 2d 54 65 73 74 25  nroll\ER NW-Test%
0060 32 30 43 41 25 32 30 54 72 61 69 6e 69 6e 67 2e  20CA%20T raining.
0070 63 72 6c 30 81 d4 06 08 2b 06 01 05 05 07 01 01  cr10... +.....
0080 04 81 c7 30 81 c4 30 5f 06 08 2b 06 01 05 05 07  ..0..0_ .+.....
0090 30 02 86 53 68 74 74 70 3a 2f 2f 77 32 6b 2e 6d  0..Shttp ://w2k.m
00a0 6f 7a 69 6c 6c 61 2e 6c 6f 63 61 6c 2f 43 65 72  ozilla.l ocal\Cer
00b0 74 45 6e 72 6f 6c 6c 2f 77 32 6b 2e 4d 6f 7a 69  tEnroll/ w2k.Mozi
00c0 6c 6c 61 2e 6c 6f 63 61 6c 5f 45 52 4e 57 2d 54  lla.local_LERNW-T
00d0 65 73 74 25 32 30 43 41 25 32 30 54 72 61 69 6e  est%20CA %20Train
00e0 69 6e 67 2e 63 72 74 30 61 06 08 2b 06 01 05 05  ing.crt0 a_+...
00f0 07 30 02 86 55 66 69 6c 65 3a 2f 2f 5c 5c 77 32  .0..ufile://\w2
0100 6b 2e 4d 6f 7a 69 6c 6c 61 2e 6c 6f 63 61 6c 5c  k.Mozilla.local\
0110 43 65 72 74 45 6e 72 6f 6c 6c 5c 77 32 6b 2e 4d  CertEnroll\w2k.M
0120 6f 7a 69 6c 6c 61 2e 6c 6f 63 61 6c 5f 45 52 4e  ozilla.l ocal ERN
  
```

C:\Daten\MyExploits\NAC\Reversing and Docs\nac-captures\healthy-1\2

```

80 12 00 18 0A FD 95 DA EF DE 12 77 80 03  . . . . .ý"Üip.w.
00 0C EF DE 12 77 1E 26 BD 57 3D 6E 14 B1 80  . . . . .ip.w.&#xW=n.±
01 00 04 F6 D3 27 A3  . . . . .ö'ë
  
```

C:\Daten\MyExploits\NAC\Reversing and Docs\nac-captures\healthy-1\3

```

00 13 00 19 0A FD 95 DB F6 D3 27 A3 80 03  . . . . .ý"Üö'ë-.
00 0C EF DE 12 77 1E 26 BD 57 3D 6E 14 B1 80 02  . . . . .ip.w.&#xW=n.±-
00 05 01 01 00 05 01  . . . . .
  
```



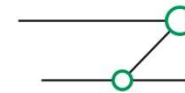
RE of the CTA – 1: Used Crypto

Address	Length	Type	String
"..." rdata:1...	0000000E	C	FIPS routines
"..." rdata:1...	0000000E	C	OCSP routines
"..." rdata:1...	00000010	C	engine routines
"..." rdata:1...	00000004	C	func(%lu)
"..." rdata:1...	00000009	C	lib(%lu)
"..." rdata:1...	0000001C	C	.\crypto\engine\tb_digest.c
"..." rdata:1...	0000001B	C	.\crypto\engine\eng_init.c
"..." rdata:1...	00000029	C	Stack part of OpenSSL 0.9.7g 11 Apr 2005
"..." rdata:1...	00000017	C	.\crypto\stack\stack.c
"..." rdata:1...	00000019	C	.\crypto\buffer\buffer.c
"..." rdata:1...	00000027	C	RSA part of OpenSSL 0.9.7g 11 Apr 2005
"..." rdata:1...	00000017	C	.\crypto\rsa\rsa_lib.c

Used crypto (btw: this version is vulnerable)



Function Hooking into EapTlvHandlePacket



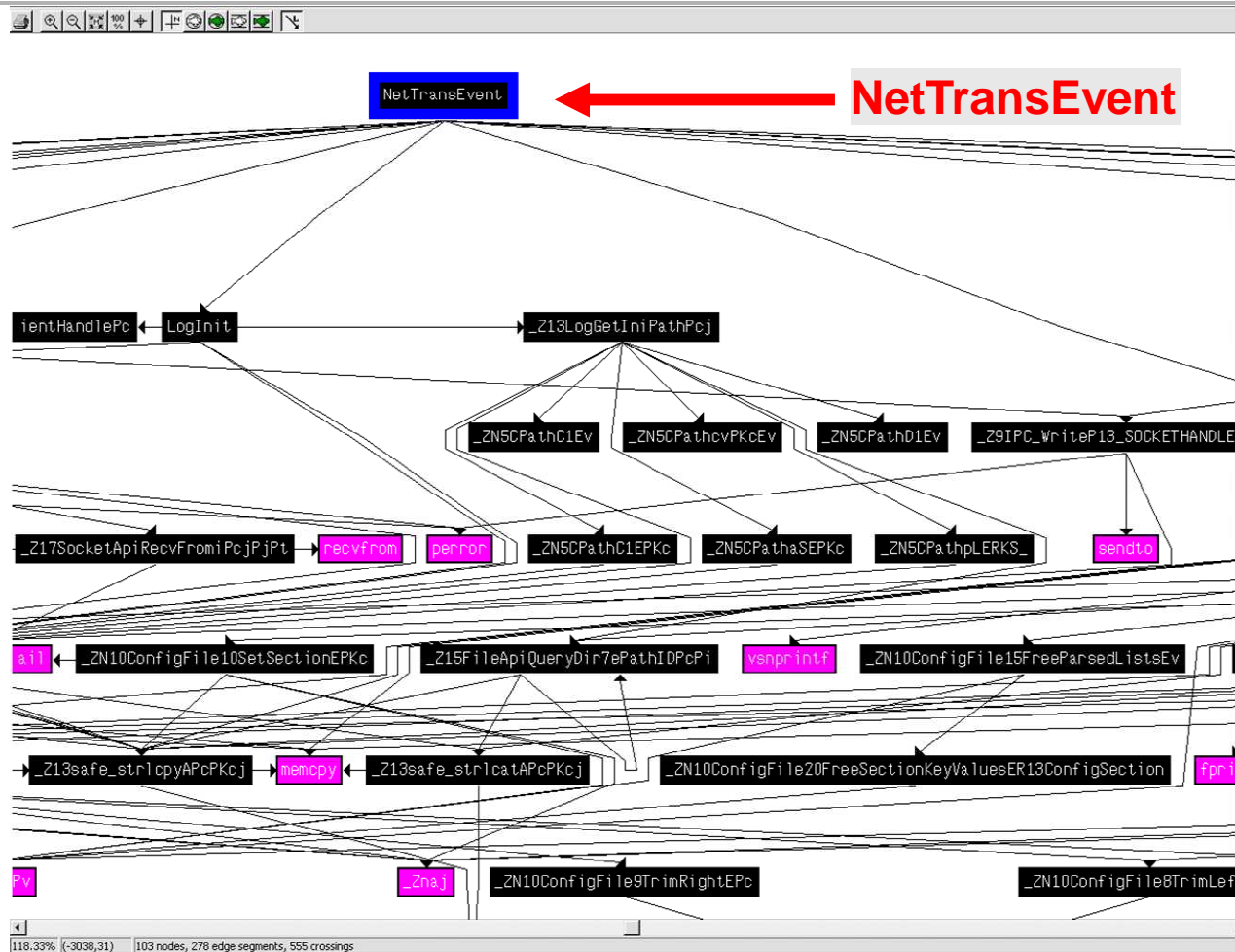
The screenshot shows the Auto Debug for Windows V4.0 interface. The left pane displays a trace of function calls, including `EapTlvHandlePacket` from `clavstlv.dll`. The right pane shows the source code of the `EapTlvHandlePacket` function, which is a hooking function. The function signature is `EapTlvHandlePacket : 0x00001048`. The parameters are listed as follows:

- param[0] = 0x00605CD8
- param[1] = 0x00607525
- param[2] = 0x00000014
- param[3] = 0x00917C1D VM-XP1-NOCTA:droecher
- param[4] = 0x00917ACC
- param[5] = 0x00607539 u=□@A=□□M\$W)@yjp%_ [K0cúTÝ□
- param[6] = 0x00927C98

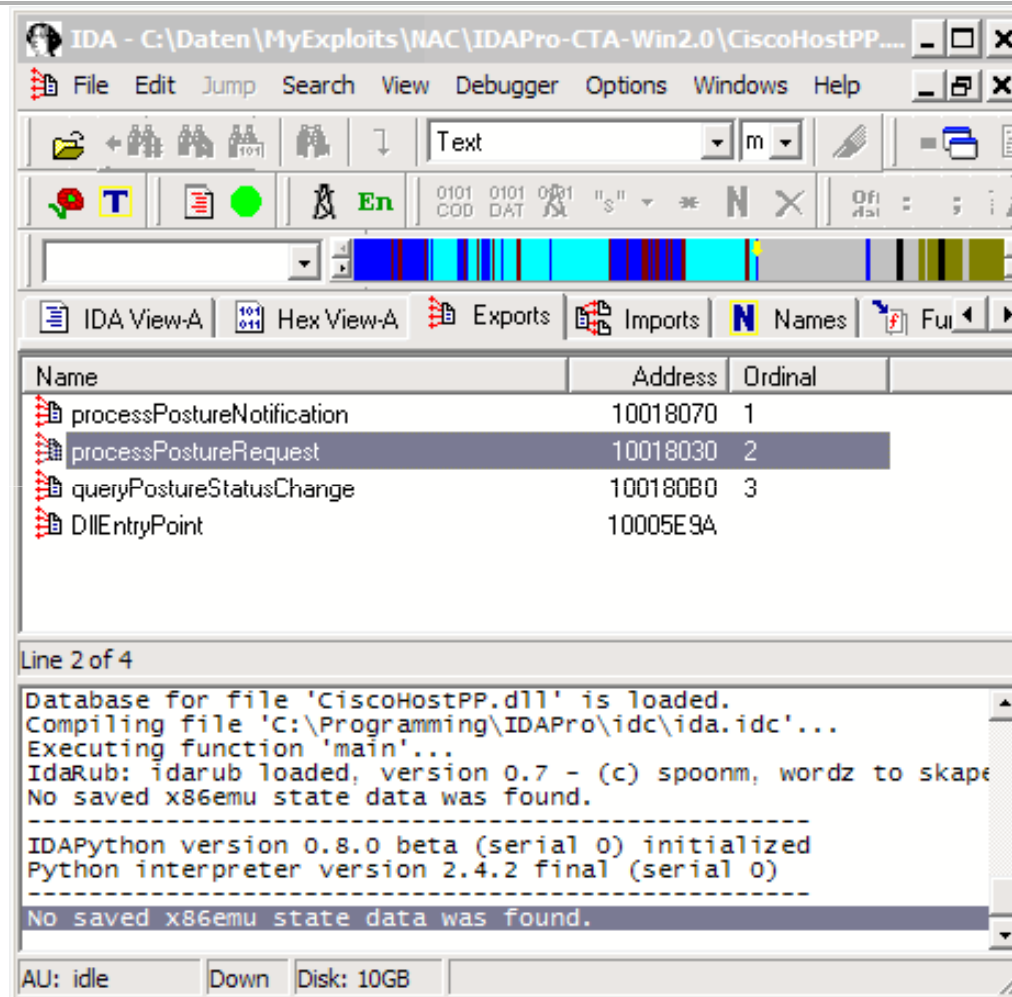
The return address is `0x007445FD`. The return value is `return = 1 (0x00000001)`. The function body shows a series of instructions, including `return 1` at the end.



RE of CTA – 3: Core Function



RE of Plug-In 1: Exported Functions



The screenshot shows the IDA Pro interface with the 'Exports' window open. The window displays a list of exported functions with their names, addresses, and ordinals. The function 'processPostureRequest' is highlighted. Below the list, the console window shows the following output:

```
Line 2 of 4  
Database for file 'CiscoHostPP.dll' is loaded.  
Compiling file 'C:\Programming\IDAPro\idc\ida.idc'...  
Executing function 'main'...  
IdaRub: idarub loaded, version 0.7 - (c) spoonm, wordz to skape  
No saved x86emu state data was found.  
-----  
IDAPython version 0.8.0 beta (serial 0) initialized  
Python interpreter version 2.4.2 final (serial 0)  
-----  
No saved x86emu state data was found.
```



RE of Plug-In 2: Exported Functions

```

; Exported entry 2. processPostureRequest

; int __cdecl processPostureRequest(char *pRequest,int ID,char *pAttributeList,int *pNumber)
public processPostureRequest
processPostureRequest proc near

pRequest= dword ptr 4
ID= dword ptr 8
pAttributeList= dword ptr 0Ch
pNumber= dword ptr 10h

mov     eax, dword_1002788C
push   esi
mov     ecx, [eax+8]
mov     edx, [eax+4]
push   ecx
push   edx
call   sub_10018000
mov     edx, [esp+0Ch+pNumber]
add     esp, 8
mov     ecx, dword_1002788C
push   edx
mov     edx, [esp+8+pAttributeList]
mov     eax, [ecx]
push   edx
mov     edx, [esp+0Ch+ID]
push   edx
mov     edx, [esp+10h+pRequest]
push   edx

; const processPostureRequest::`vftable'
??_7processPostureRequest@@6B@:
call   dword ptr [eax+4]
mov     esi, eax
call   sub_10018020
mov     eax, esi
pop     esi
retn

processPostureRequest endp

; Exported entry 1. processPostureNotification

; int __cdecl processPostureNotification(char *NotifyBuffer,int Status)
public processPostureNotification
processPostureNotification proc near

NotifyBuffer= dword ptr 4
Status= dword ptr 8

mov     eax, dword_1002788C
push   esi
mov     ecx, [eax+8]
mov     edx, [eax+4]
push   ecx
push   edx
call   sub_10018000
mov     edx, [esp+0Ch+Status]
mov     ecx, dword_1002788C
add     esp, 8
mov     eax, [ecx]
push   edx
mov     edx, [esp+8+NotifyBuffer]
push   edx
push   edx
call   dword ptr [eax+8]
mov     esi, eax
call   sub_10018020
mov     eax, esi
pop     esi
retn

processPostureNotification endp

; Exported entry 3. queryPostureStatusChange

; int __cdecl queryPostureStatusChange()
public queryPostureStatusChange
queryPostureStatusChange proc near

mov     eax, dword_1002788C
push   esi
mov     ecx, [eax+8]
mov     edx, [eax+4]
push   ecx
push   edx
call   sub_10018000
mov     ecx, dword_1002788C
add     esp, 8
mov     eax, [ecx]
call   dword ptr [eax+0Ch]
mov     esi, eax
call   sub_10018020
mov     eax, esi
pop     esi
retn

queryPostureStatusChange endp

```



Quick Summary...

- **A lot of stuff learned so far...**
 - What is used
 - How it works
 - How it interoperates
 - Where to start hacking it

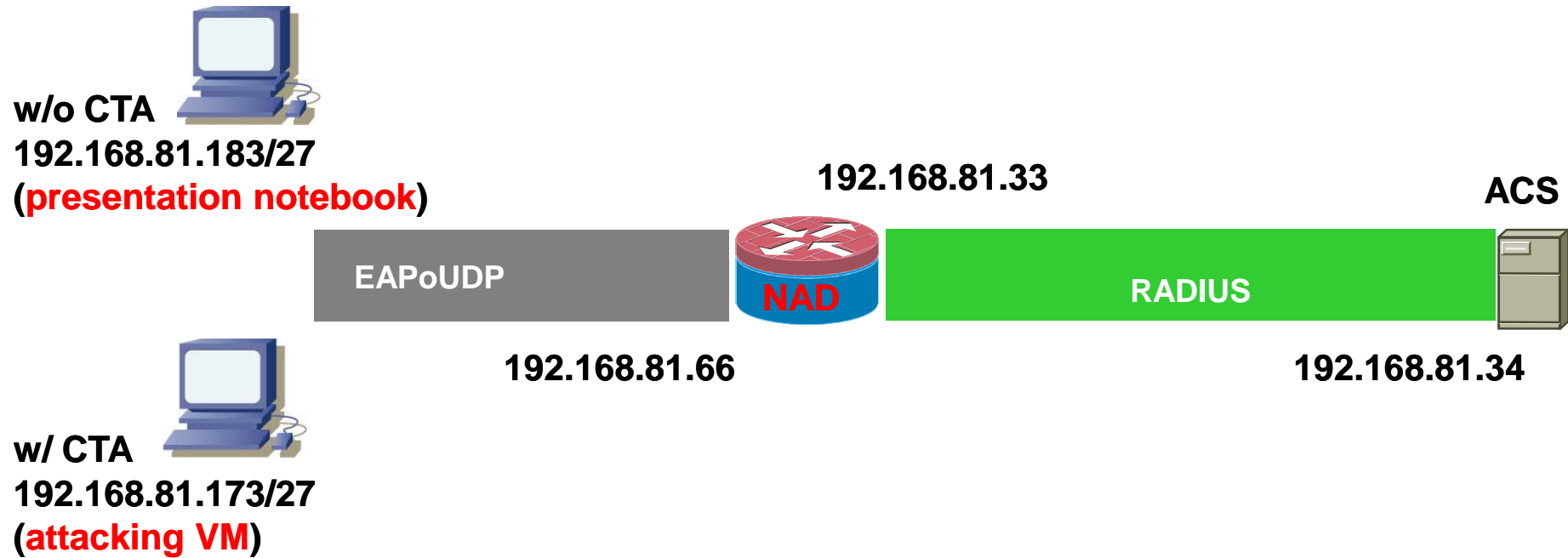
- **So now its...**



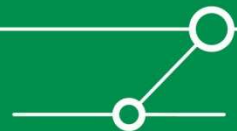
SHOWTIME



Showtime Setup



Part 5 – Some thoughts on mitigation



Mitigation isn't just a "patch"

- **As we have shown the problems are related to design-flaws.**
- **We have shown that these are serious – we consider Cisco NAC to be “hacked” in its current version.**
- **Problem is: A simple patch won't solve the issue. It's not like a “software problem” related to a BO. It's a design-problem (as e.g. in WEP).**



Mitigation by Cisco -1: Code Signing

- **Code Signing** the plug-ins and running only signed plug-ins from a trusted source would defeat plug-in replacement attacks.
- We can not judge the effort needed to implement code signing but we would heartily welcome seeing signed code in any (security related) product.



Mitigation by Cisco – 2: Mandatory Authentication

- **Strong mandatory client-authentication** would stop outsider attacks against the NAC framework. Adding authentication (mandatory or, in a first step, optional) should be possible without too much of a change as PEAP is being used and PEAP has built-in authentication capabilities.
- The reasons for not having authentication in the framework can only be business-related – Cisco knows that implementing NAC is already a major effort and probably does not want to put additional stress on its clients by making authentication mandatory.



By the Customer 1: Strong Authentication

- **Strong Authentication:** Whenever possible 802.1x-based NAC should be implemented in order to add strong authentication to the authorization process.
- If 802.1x is not feasible, other means of strong authentication should be implemented.
- In RAS-VPN scenarios for example, where NAC-Layer3-IP is the only NAC-flavor available, clients should be subjected to strong authentication on the VPN-device itself.
- The “strong authentication” mitigates threats posed by the “outside attacker”.



By the Customer 2: Least Privilege

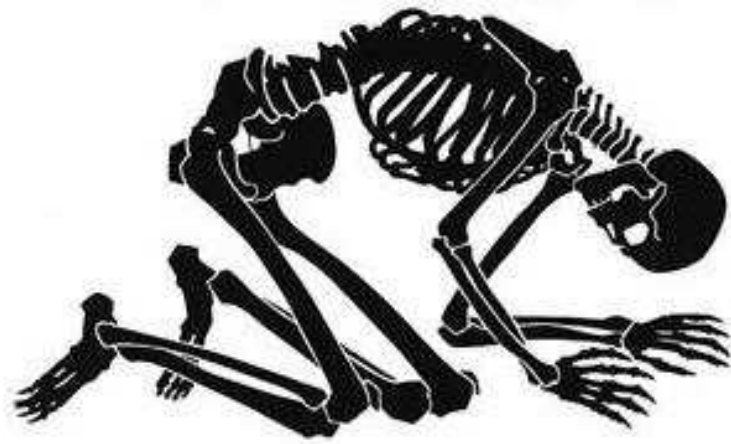
- **Least Privilege:** All attack-vectors for “inside attackers” have a common characteristic. They need “tampering” with the CTA installation.
- In case of “plug-in”-replacement the authentic plug-ins are being replaced by self-written plug-ins.
- A possible mitigation could be to enforce strict access-rights on the plug-in files by ensuring that users don't have administrative privileges.
- In case of “alternative client” “file access restrictions” is not a possible mitigating control.



By the Customer 3: CSA

- **CSA instead of CTA:** In addition to the CTA Cisco also offers a host based IDS in the name of “Cisco Security Agent” which also includes the CTA (in some versions) and has its own CTA plug in.
- The CSA monitors the integrity of the CTA and will prevent illegitimate changes to the CTA. This will mitigate threats posed by the “inside attacker”.
- Other HIPS normally include similar functionality but may not include a NAC plug-in.





Thank's for your patience

Time left for `questions & answers` ?

