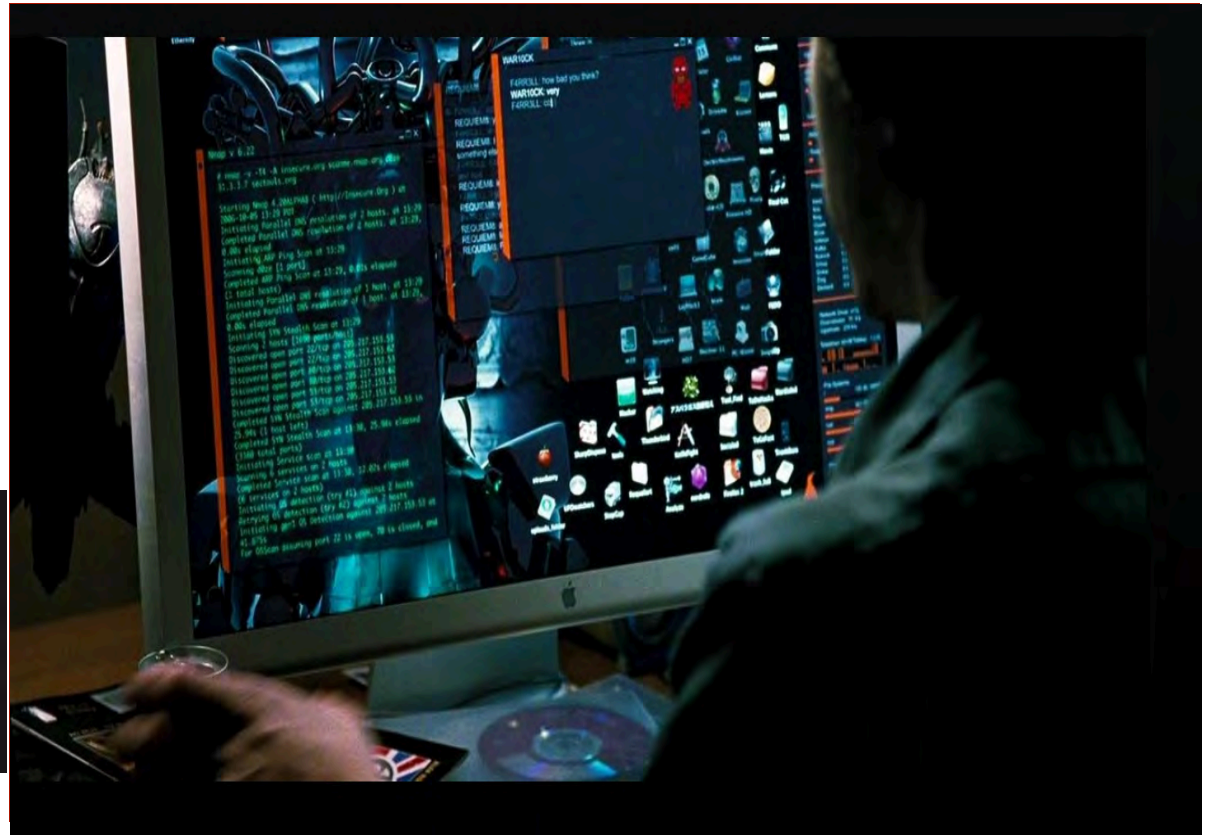


Why Nation-State Malwares Target Telco Networks:

Dissecting Technical Capabilities of Regin and Its Counterparts

Author: Ömer Coşkun

The supreme art of war is to subdue the enemy without fighting. *Sun Tzu*



Outline

1

- Overview
  - Telecom Network Architecture
  - Practical Attack Surfaces
    - GRX Attack Vectors
    - SS7 Attack Vectors
  - Practical Attack Scenarios

- Rootkit Attacks: Regin and it's counterparts
  - Common Rootkit Techniques and Regin
  - Regin vs. Uruborus and Duqu
  - ***Demo: PoC || GTFO***

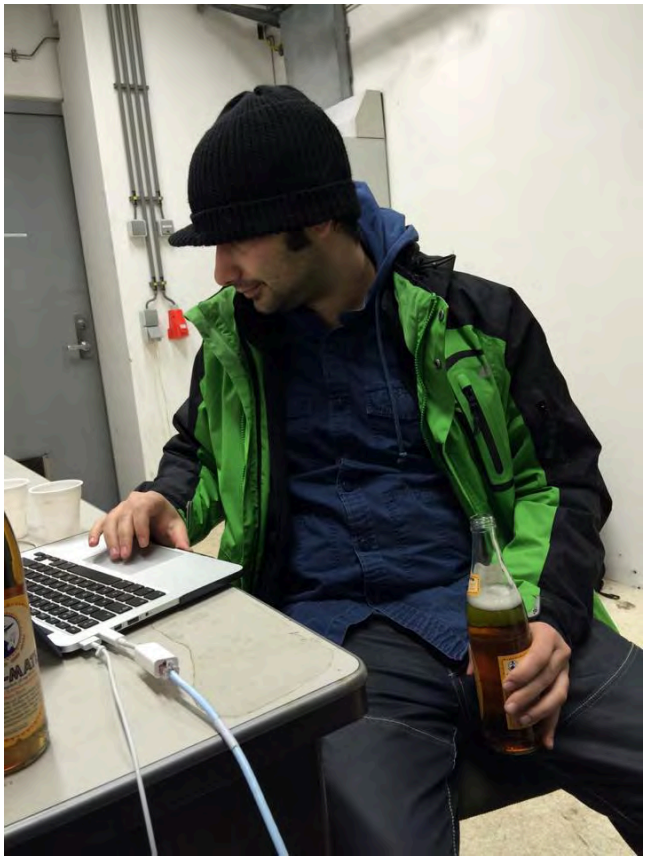
- Questions ?

kpn



$ whoami

2



Ömer Coşkun (@0xM3R)

- BEng. Computer Science

  Research Assistant in Quantum Cryptography & Advanced Topics in AI

- Industry Experience

  KPN – CISO , Ethical Hacking

  Verizon – Threat & Vulnerability Management

  IBM ISS – Threat Intelligence

- Interests

Algorithm Design, Programming, Cryptography, Reverse Engineering, Malware Analysis, OS Internals, Rootkits

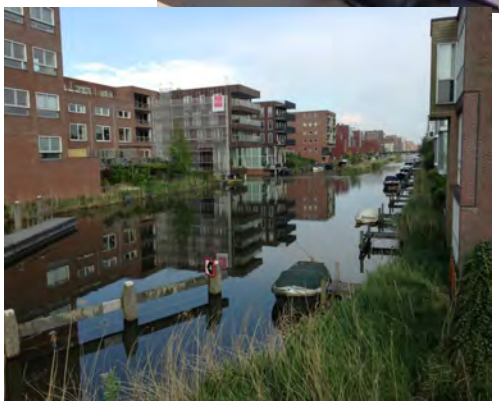
kpn

$ REDteam

3











kpn



Motivations

4

- Analyze existing vulnerabilities and attack surface of GSM networks
- Governments hack their own citizens
- Surveillance implants shifted focus to telecom networks and network devices
- European Telco companies are really paranoid after Regin attack
- Rootkits are fun : a lot to learn & challenge
- Reproduce the attack scenario and implement it!



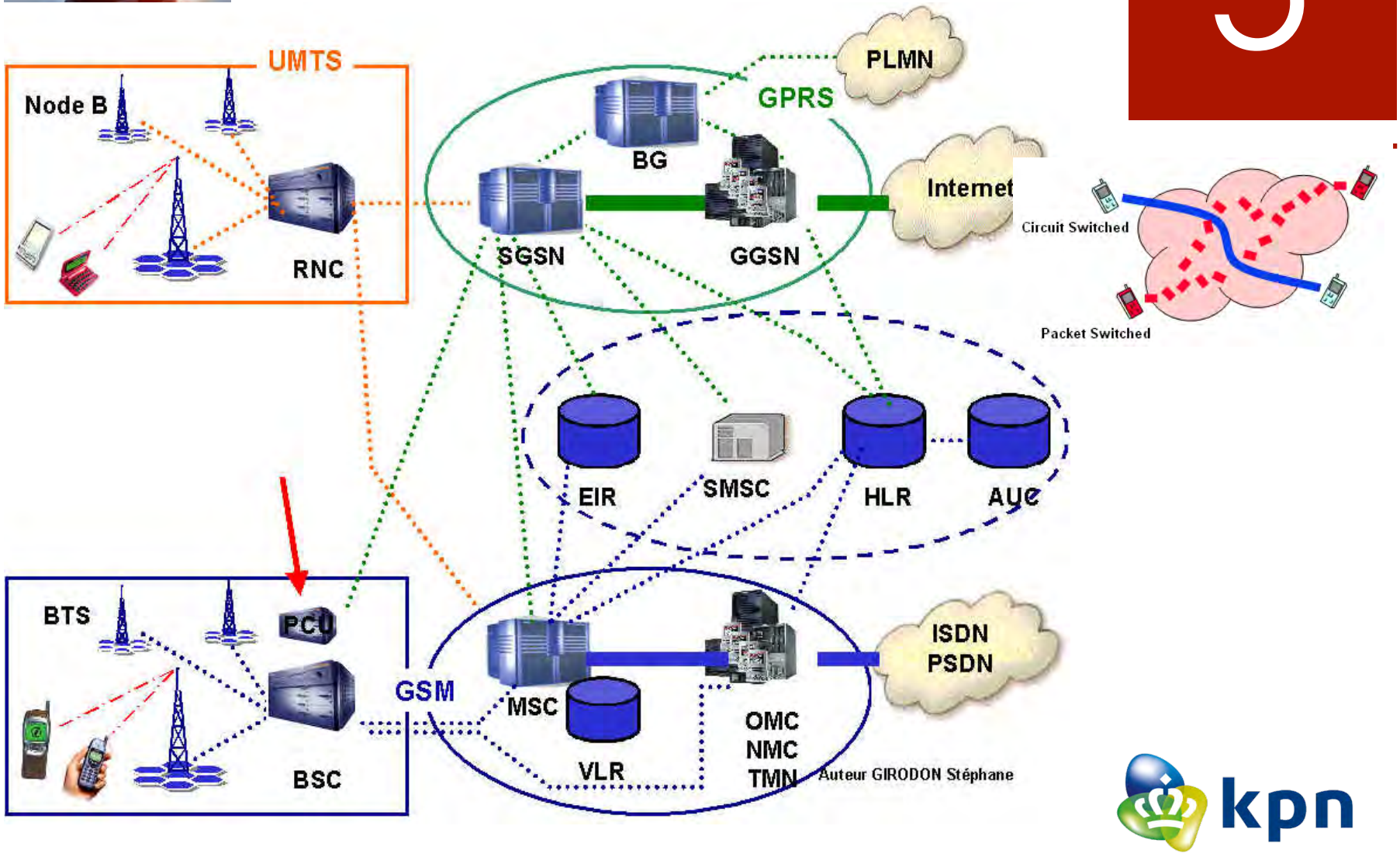


kpn



GSM Network Architecture

5

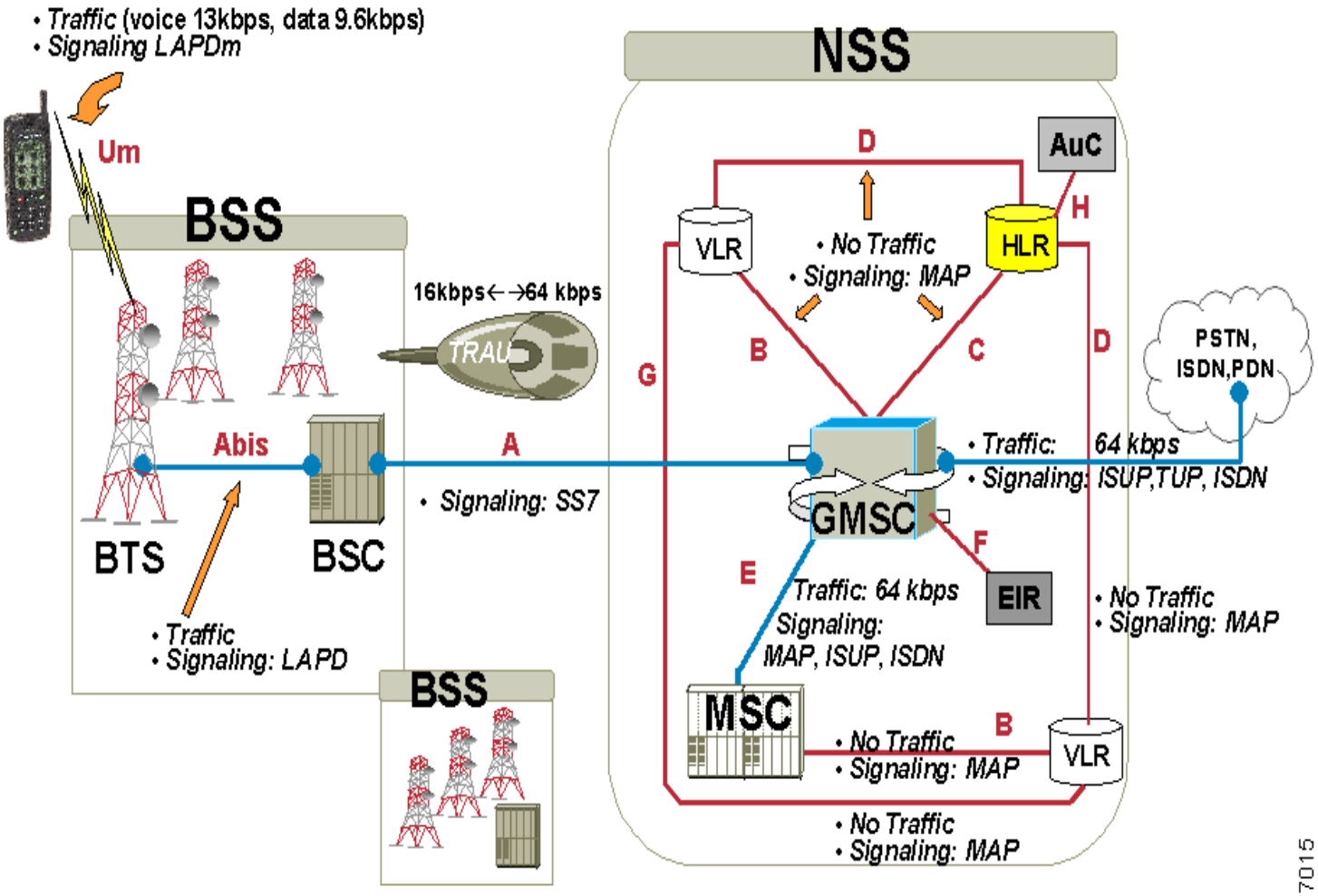


kpn



GSM Network Architecture

6



77015

kpn

Regin targets GSM Networks

7

Regin: nation-state ownage of GSM networks

"Beware of Regin, the master! His heart is poisoned. He would be thy bane..."

By GReAT on November 24, 2014. 2:00 pm

The Register
Biting the hand that feeds IT

DATA CENTRE SOFTWARE NETWORKS SECURITY INFRASTRUCTURE BUSINESS HARDWARE SCIENCE BOOTNOTES

'Regin': The 'New Stuxnet' spook-grade SOFTWARE WEAPON described

'A degree of technical competence rarely seen'



Regin Is 'Groundbreaking' Malware Used by UK Spooks

November 24, 2014 // 01:16 PM EST

MOTHERBOARD

Written by
JOSEPH COX
CONTRIBUTOR

kpn

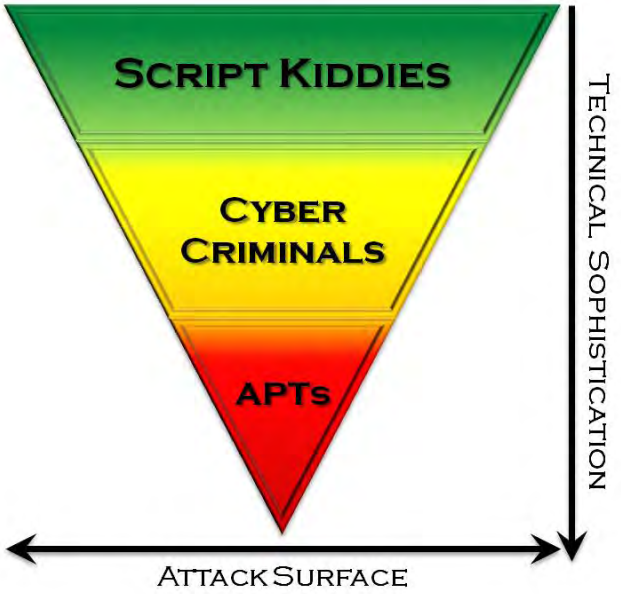


Determining Attack Surface









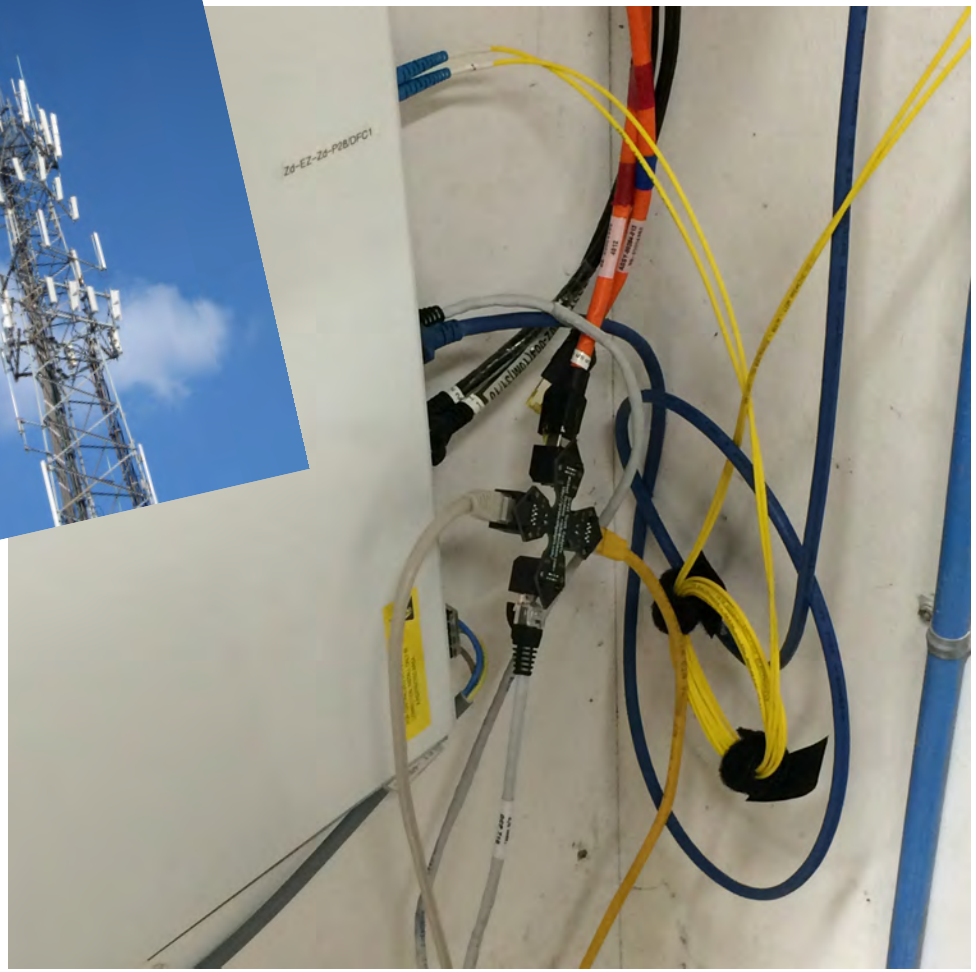
kpn



Determining Attack Surface

9







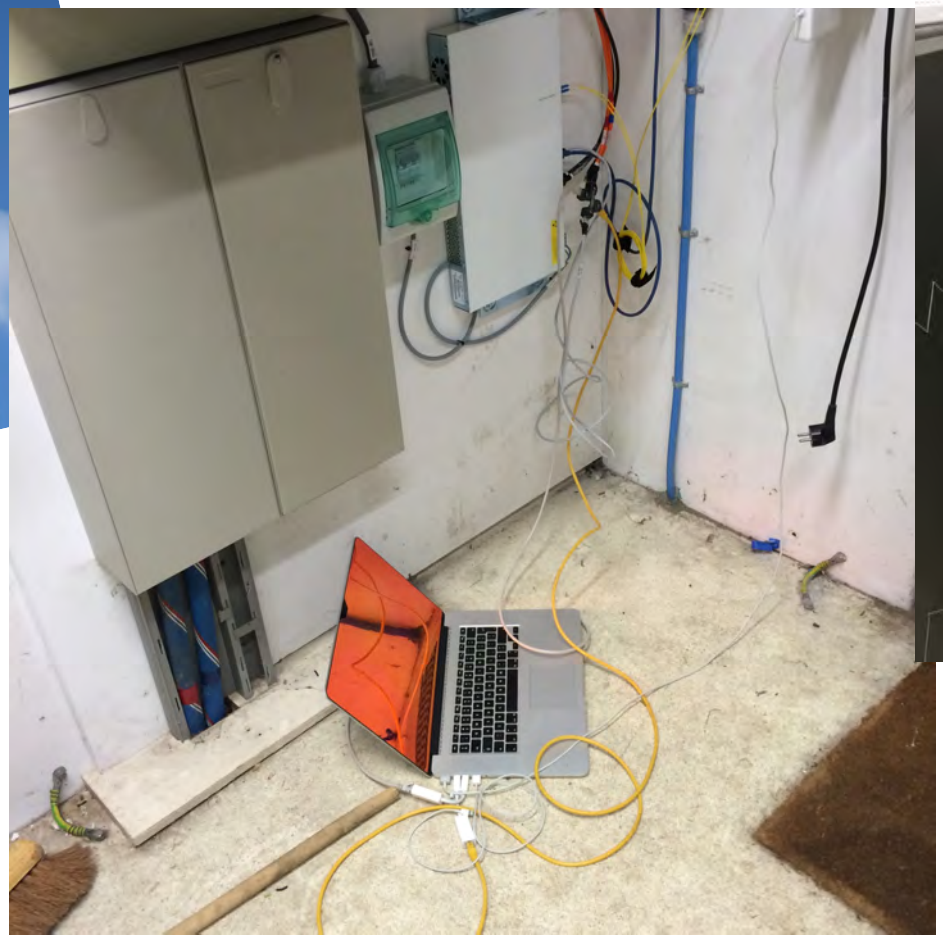
kpn



Determining Attack Surface

10







kpn



Potential Attack Surfaces

- Absence of physical intrusion detection devices
- Vulnerable services running accessible from BTS
- Absence of tamper resistance and unauthorized access protection
- Improper network segmentation; inner non-routable segments of the Telco company could accessible.
- Core GPRS Network and Network Subsystem (NSS) could be exploitable!

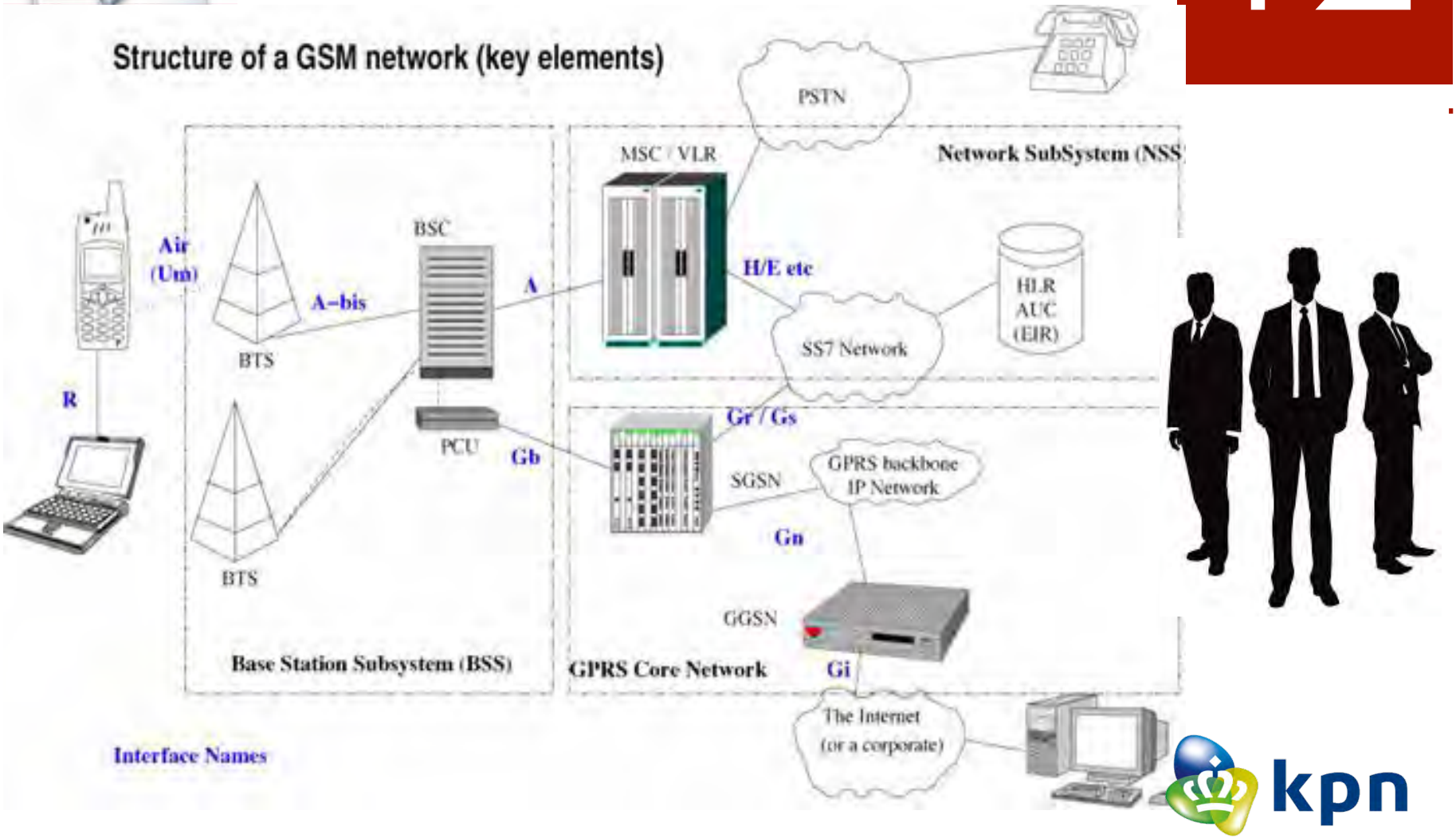


kpn

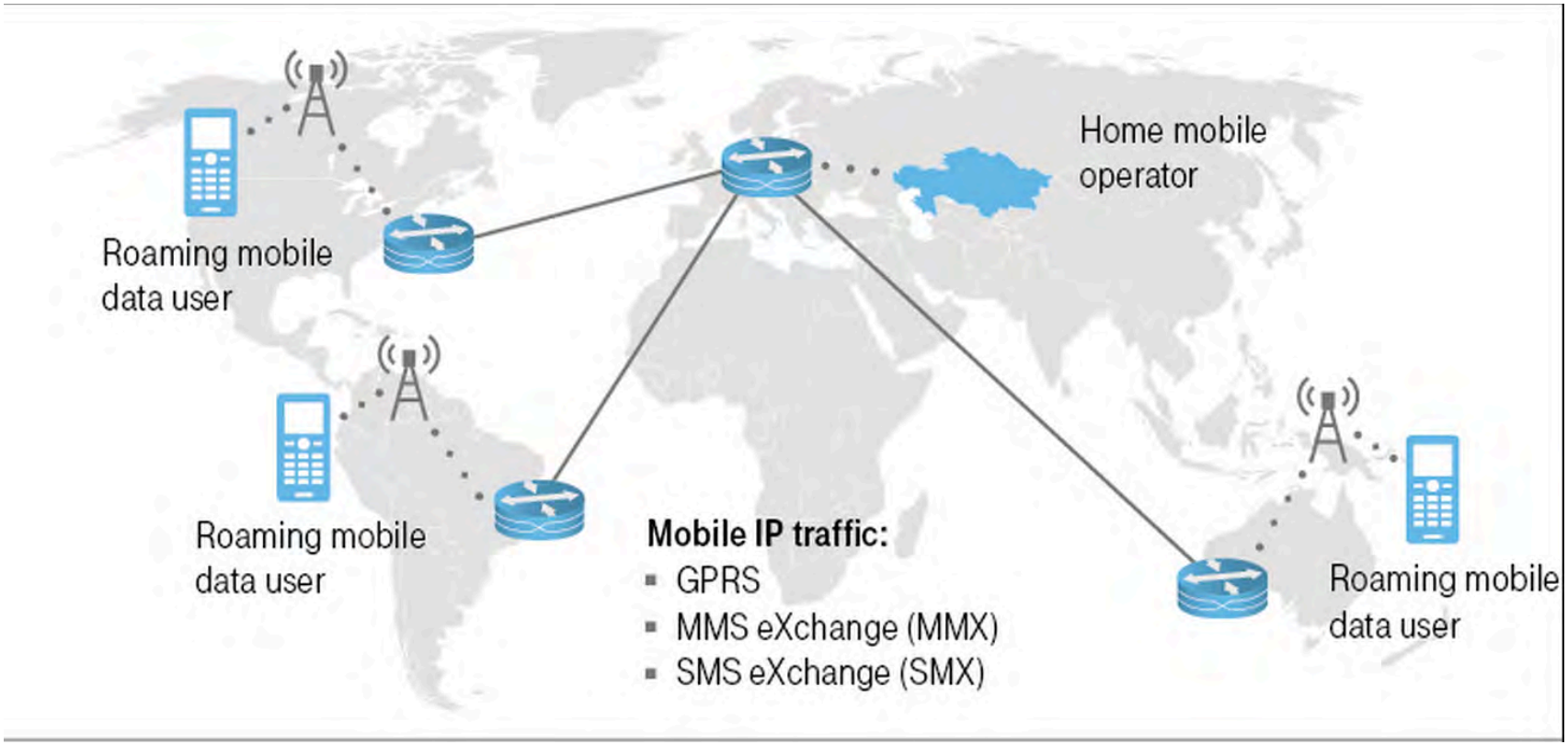


Potential Attack Surfaces

12



GRX Networks



GRX Networks

14

- GPRS roaming exchange, interconnecting networks.
- Your local GSM provider abroad
- Trust-based, highly interconnected network, made for internet sharing
- A failure or malicious activity would affect multiple connected machines
- Multiple attacks vectors, not limited to a particular segment where you are originating from.

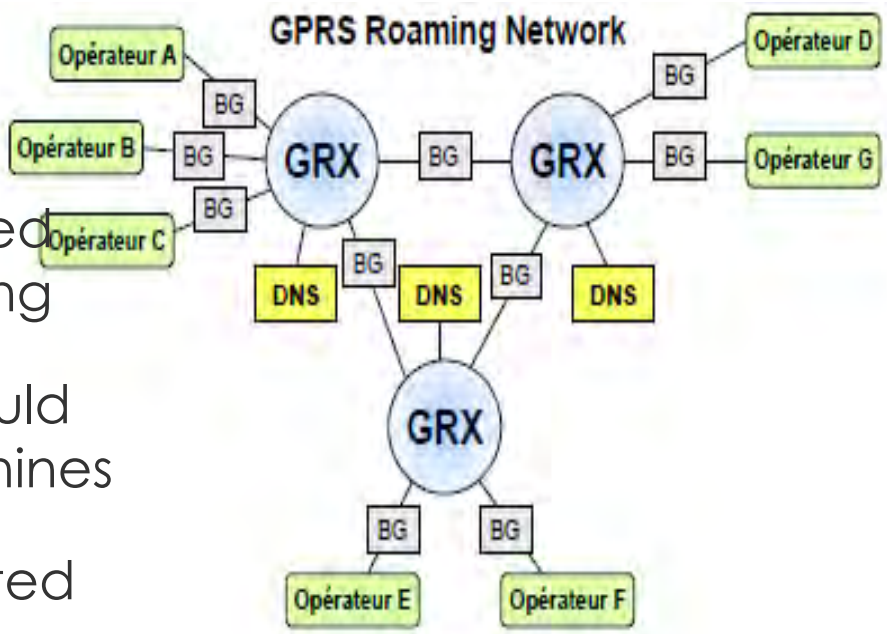
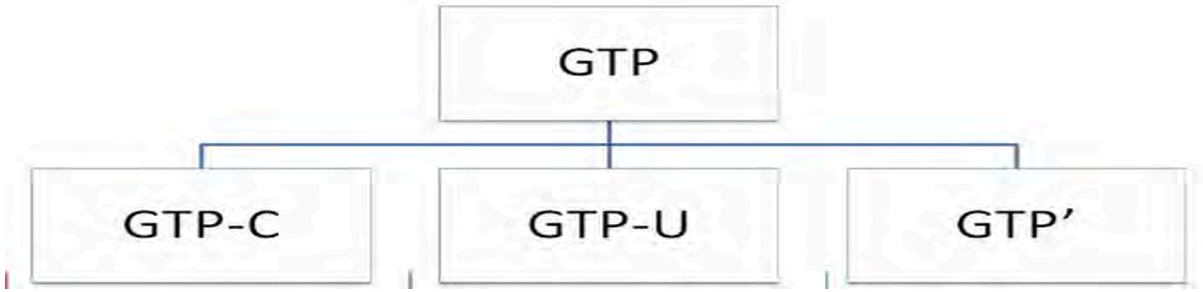
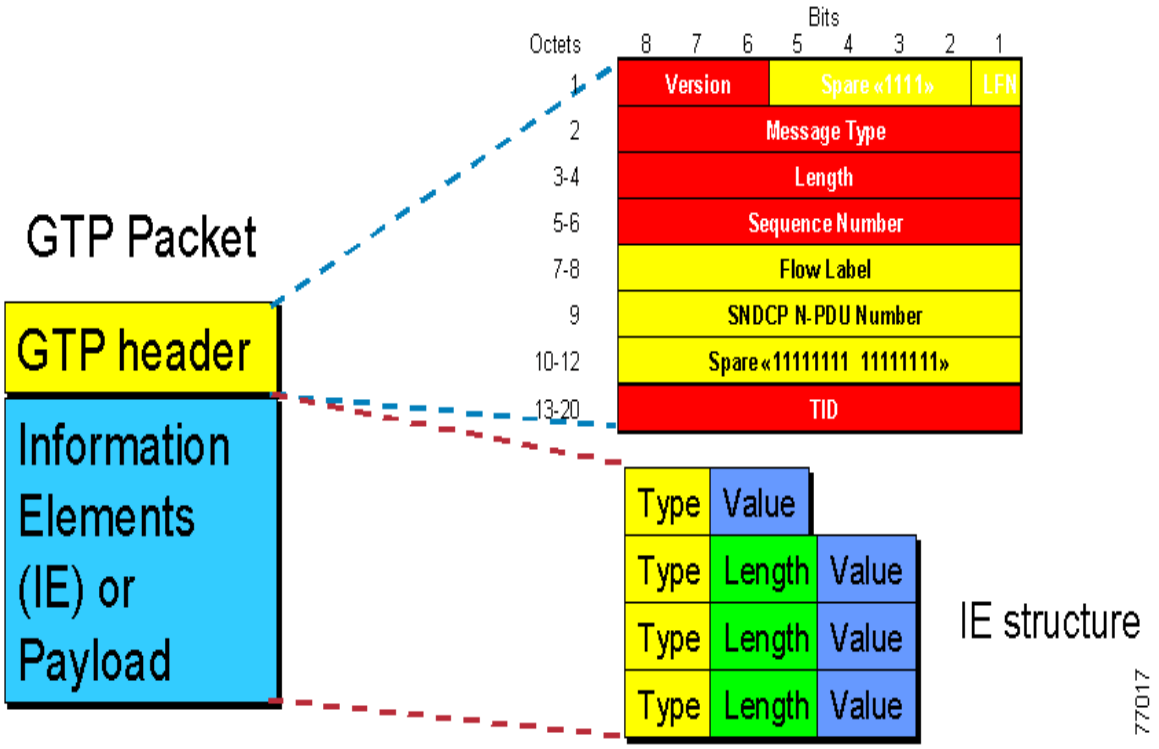


Figure 3 : Réseau GRX

GRX Networks – Attack Vectors





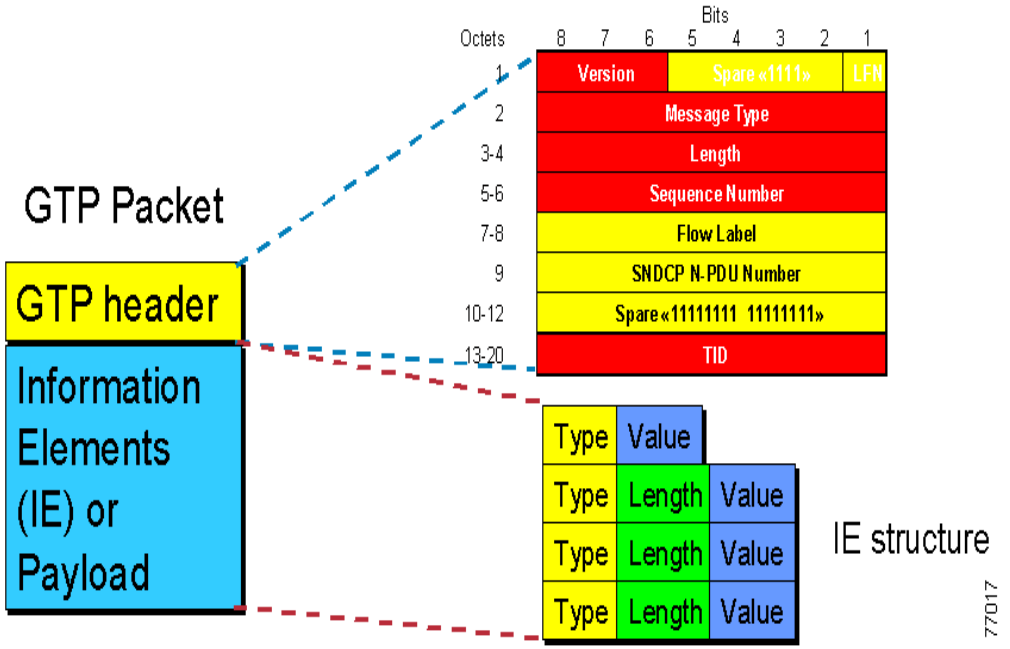
77017

kpn

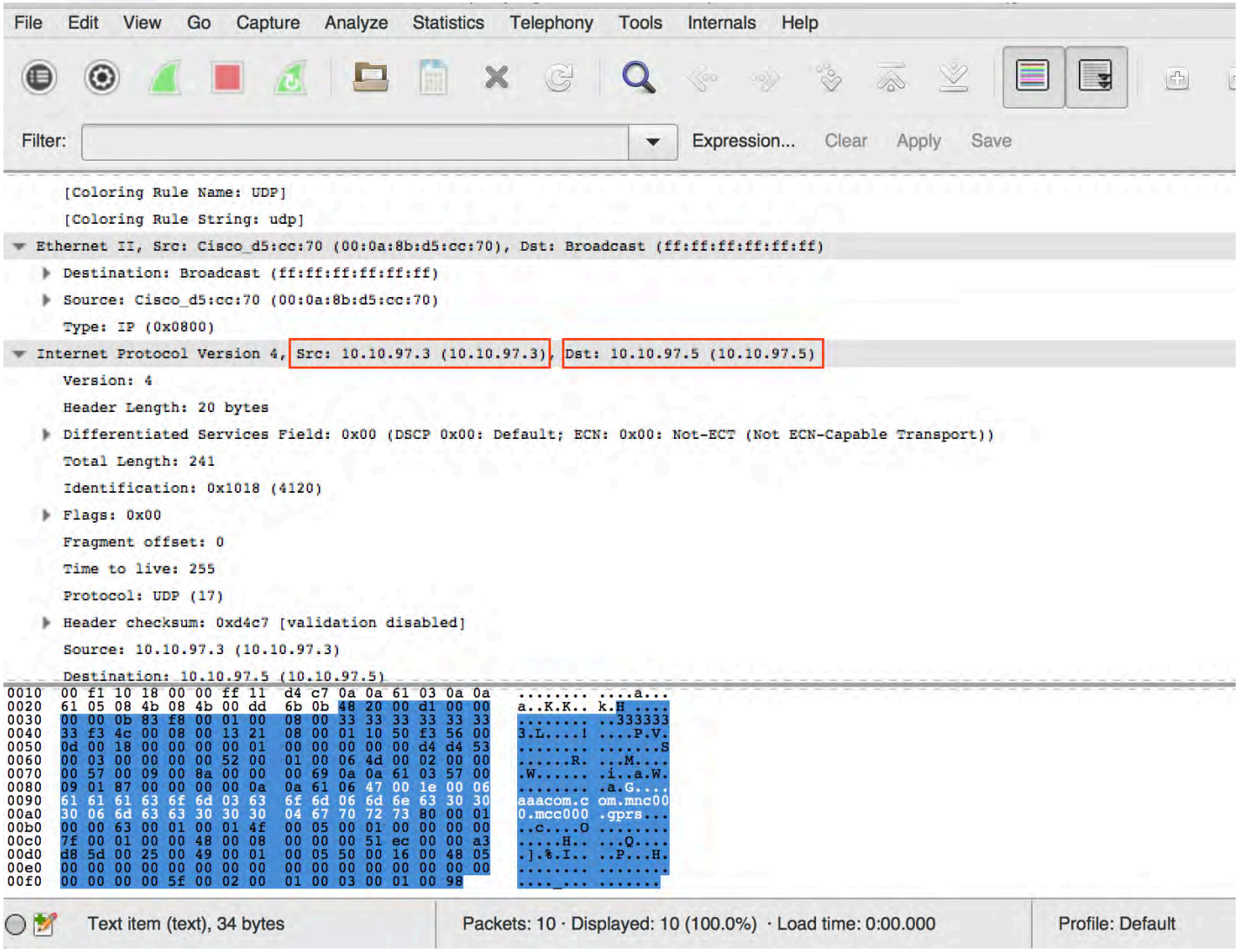
GRX Networks – Attack Vectors

16

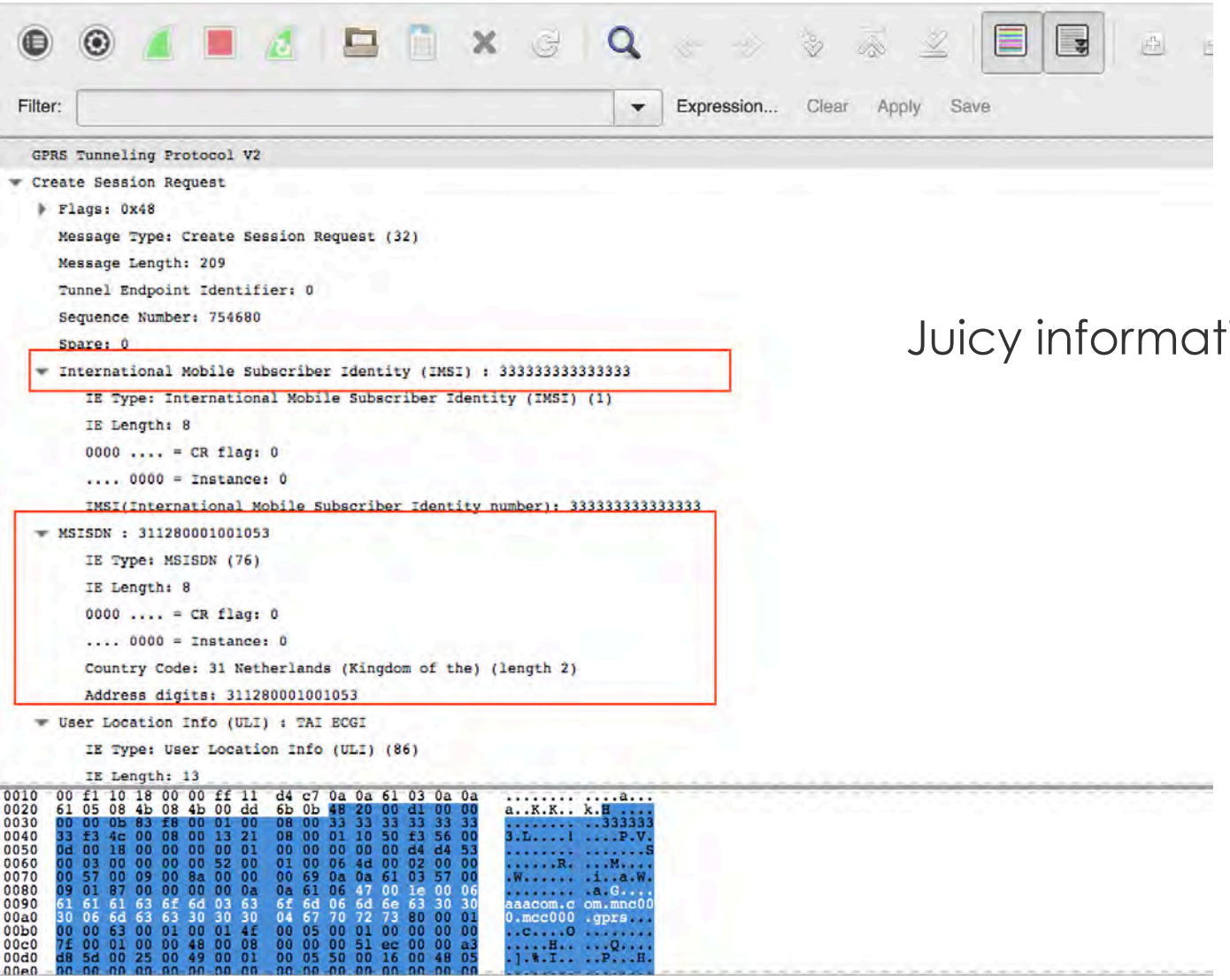
- GPRS roaming exchange, interconnecting networks.
- Your local GSM provider abroad
- Trust-based, highly interconnected network, made for internet sharing
- Multiple attacks vectors, not limited to a particular segment where you are originating from.



GRX Networks – Network Flow

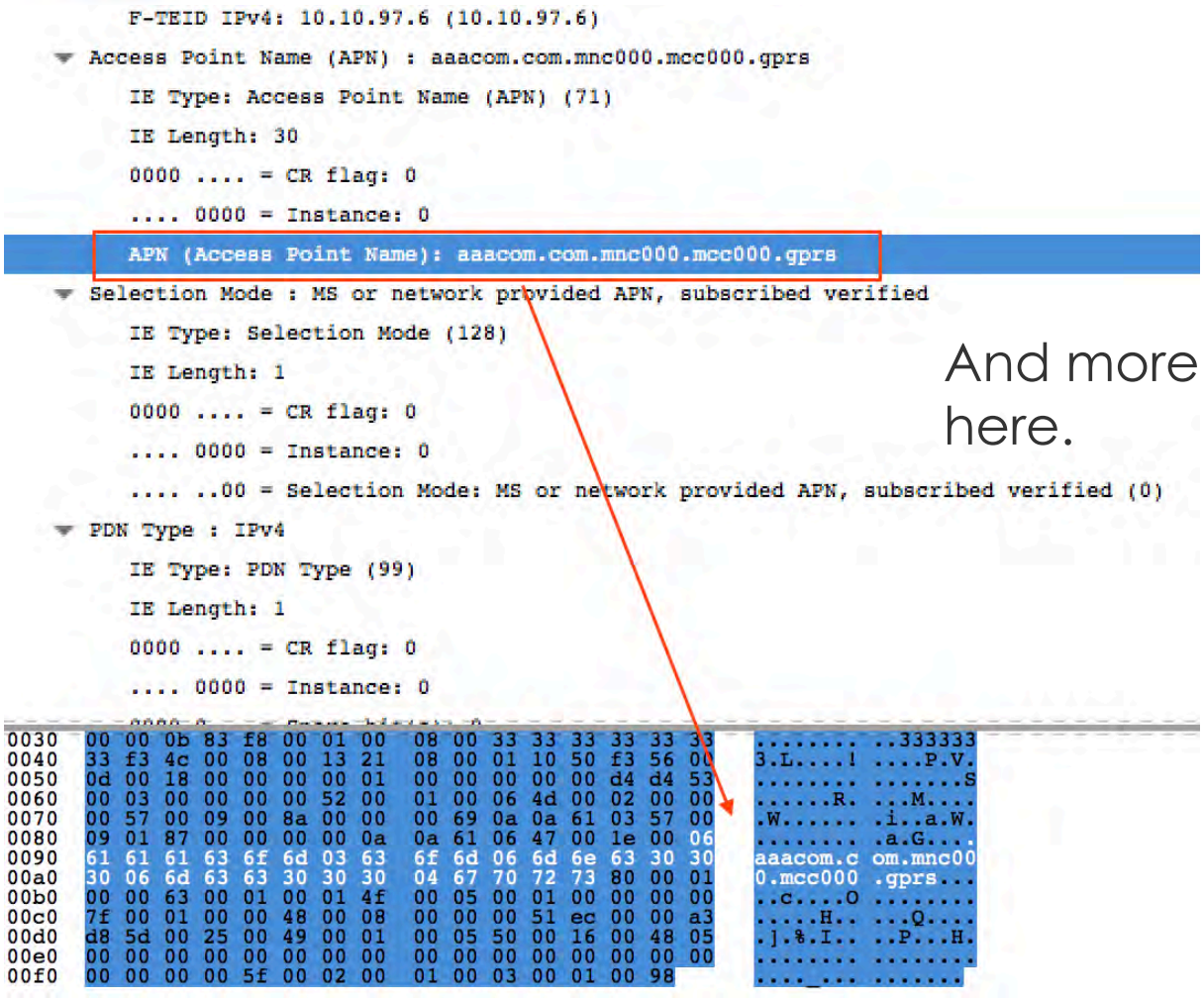


GRX Networks – Network Flow



Juicy information is here.

GRX Networks – Network Flow



And more juicy information is here.



GRX Networks – Attacks & Flaws

Are you telling me all your communication intercepted and logged including your physical location?.

NSA: We'll move your metadata into /dev/null

stop suing us

collection



The NSA promises to delete its phone metadata early next year

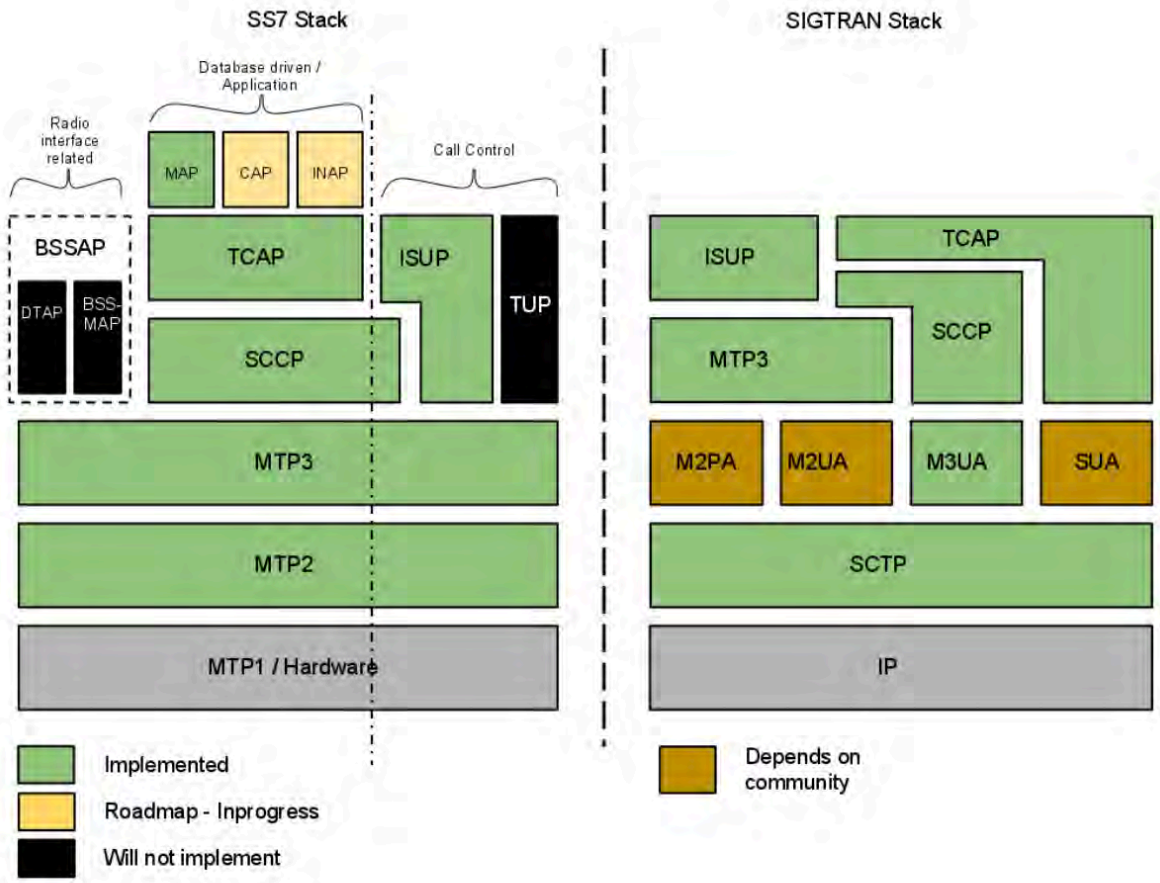
Digital Trends - 2 days ago

The **National Security Agency will** lose the power to keep and



kpn

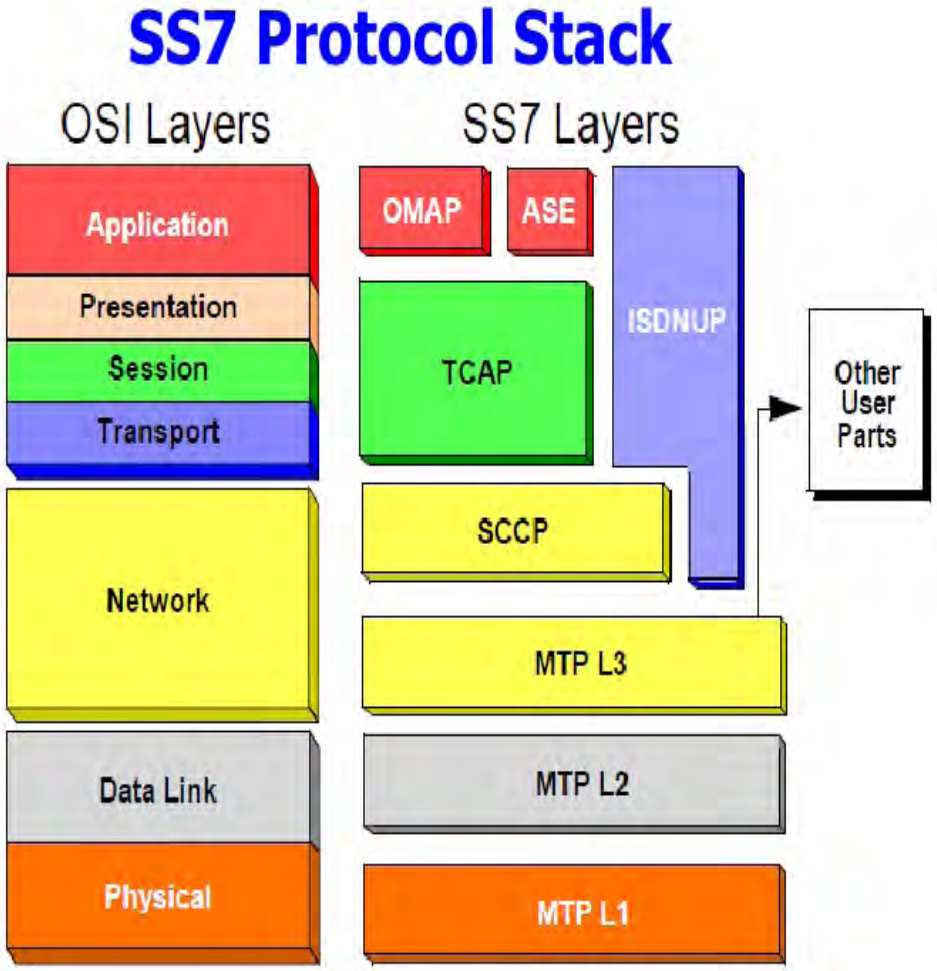
SS7 & SIGTRAN



http://www.telestax.com/

SS7 & SIGTRAN

22



SS7 Introduces procedures for

- User identification.

Routing

- Billing
- Call management

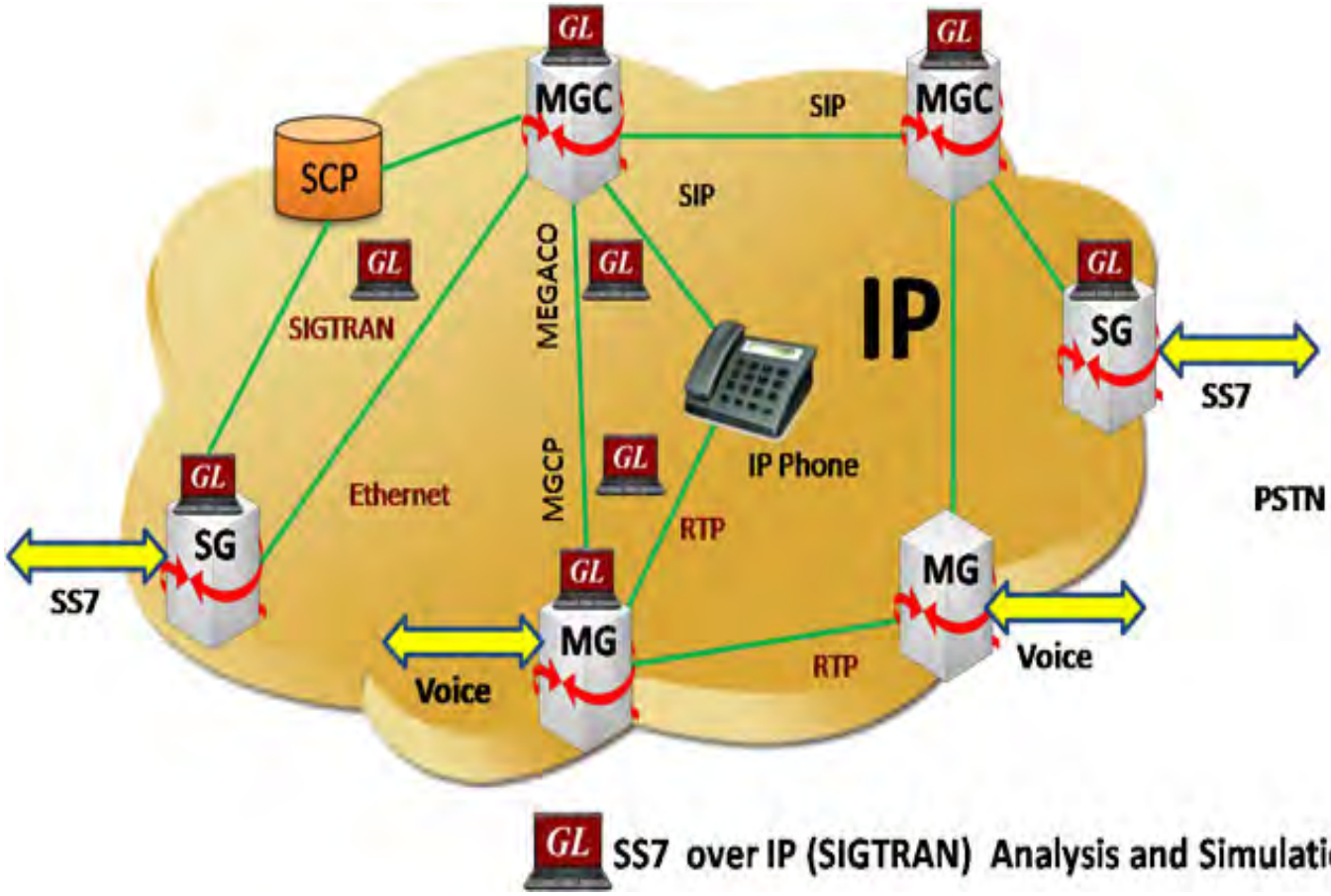
SS7 & SIGTRAN

- SS7 Features:

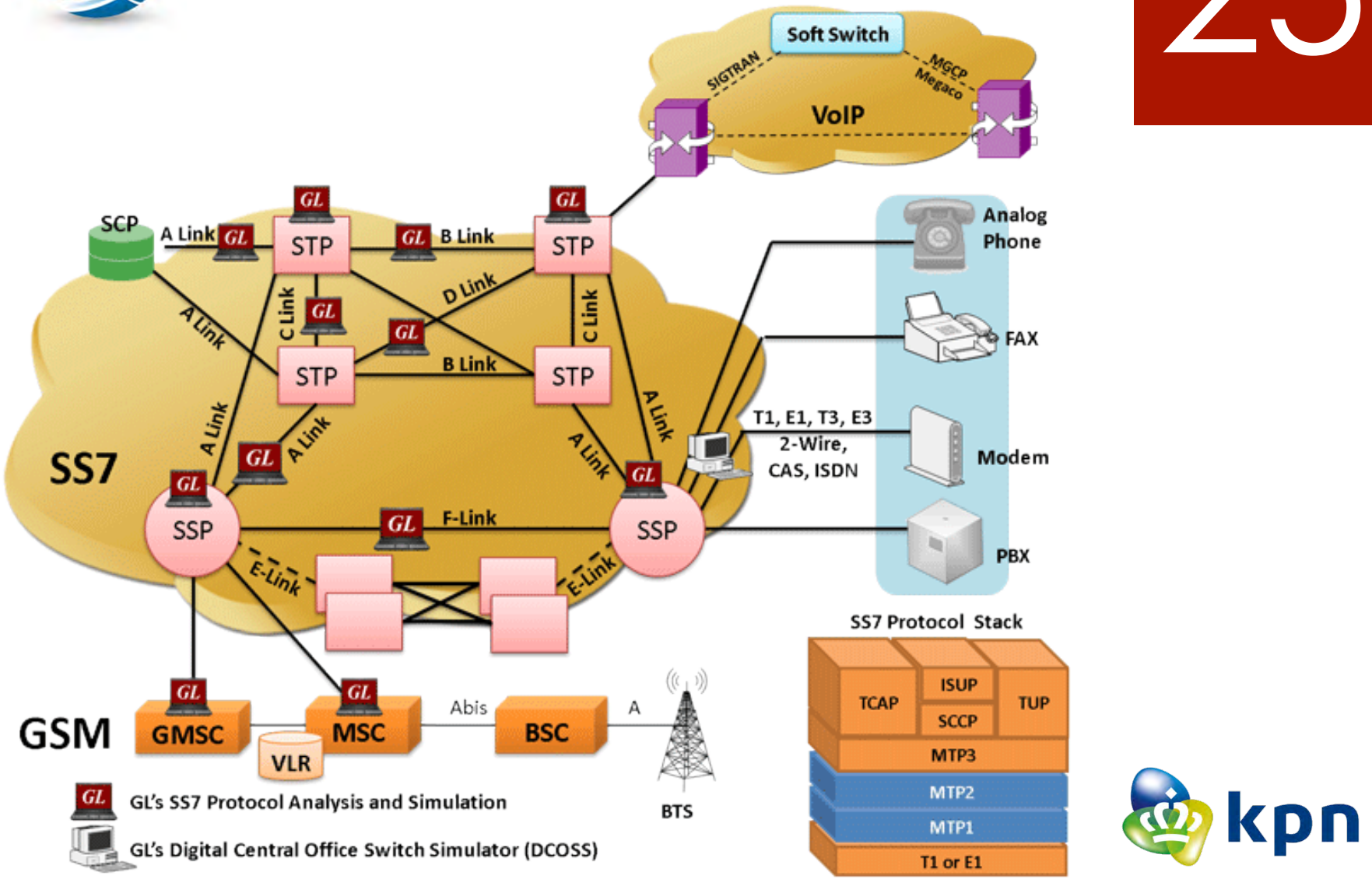
- Flow control of transmitted information
- Traffic congestion controls
- Peer entity status detection (GT + PC or SPC)
- Traffic Monitoring and monitoring measuremen



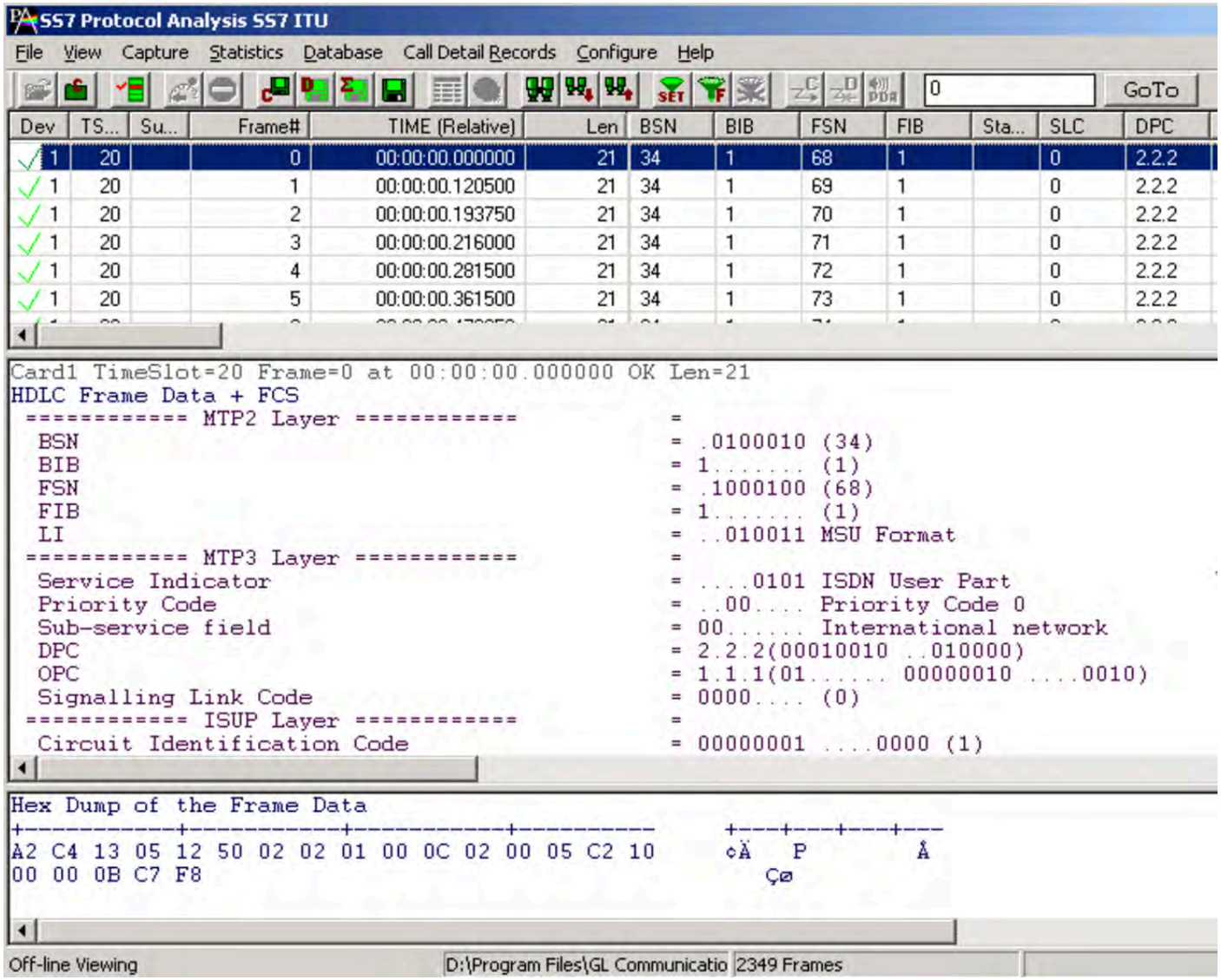
SS7 & SIGTRAN


GL
MGC
SIP
MGC
SCP
SIP
MEGACO
SIGTRAN
IP
SG
SS7
IP Phone
MGCP
Ethernet
PSTN
SG
SS7
RTP
MG
Voice
RTP
MG
Voice
GL SS7 over IP (SIGTRAN) Analysis and Simulation

SS7 & SIGTRAN

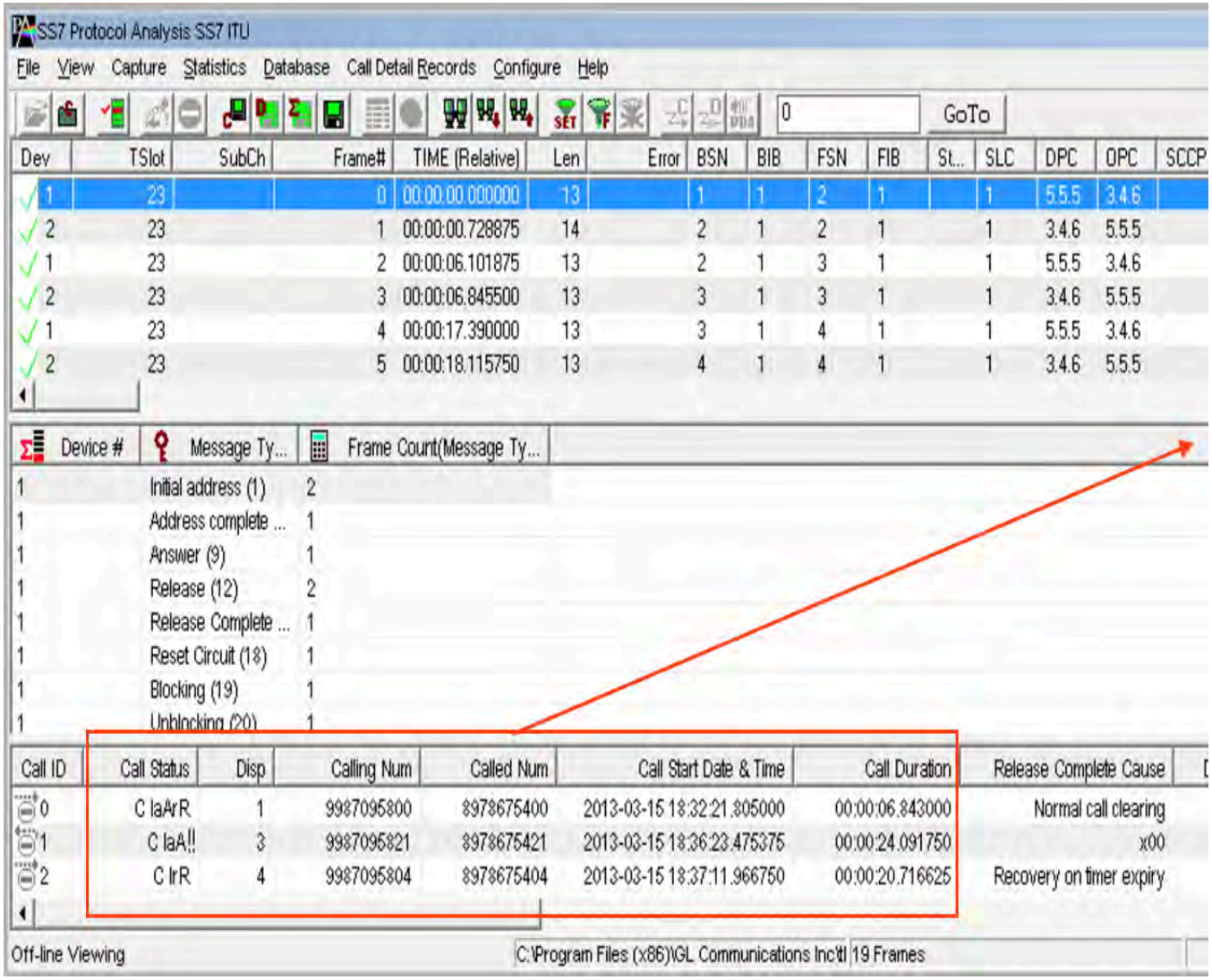

Soft Switch
SIGTRAN
MGCP
Megaco
VoIP
SCP
A Link
STP
B Link
D Link
C Link
SS7
SSP
F-Link
E-Link
Analog Phone
FAX
Modem
PBX
T1, E1, T3, E3
2-Wire,
CAS, ISDN
GSM
GMSC
MSC
VLR
Abis
BSC
A
BTS
GL's SS7 Protocol Analysis and Simulation
GL's Digital Central Office Switch Simulator (DCOSS)
SS7 Protocol Stack
TCAP
ISUP
SCCP
TUP
MTP3
MTP2
MTP1
T1 or E1

SS7 Protocol Analysis



SS7 Protocol Analysis

27



All the juicy info here :

- ✓ **Calling no.**
- ✓ **Called no**
- ✓ **Call duration**
- ✓ **Call duration**
- ✓ **Call status**

kpn



SS7 Protocol Attacks & Flows

Feel confident that NSA not interested in 'Good' people?.

NSA: We'll move your metadata into /dev/null when you stop suing us

November 29 set as cutoff date for further collection

NSA Not Interested in "Bad People"; It Listens to "Interesting ...

www.youtube.com/watch?v=BhrPVse9TnE

Feb 4, 2015 - Uploaded by Ken Campbell

Michael Hayden, former **NSA** Director, former Director of National



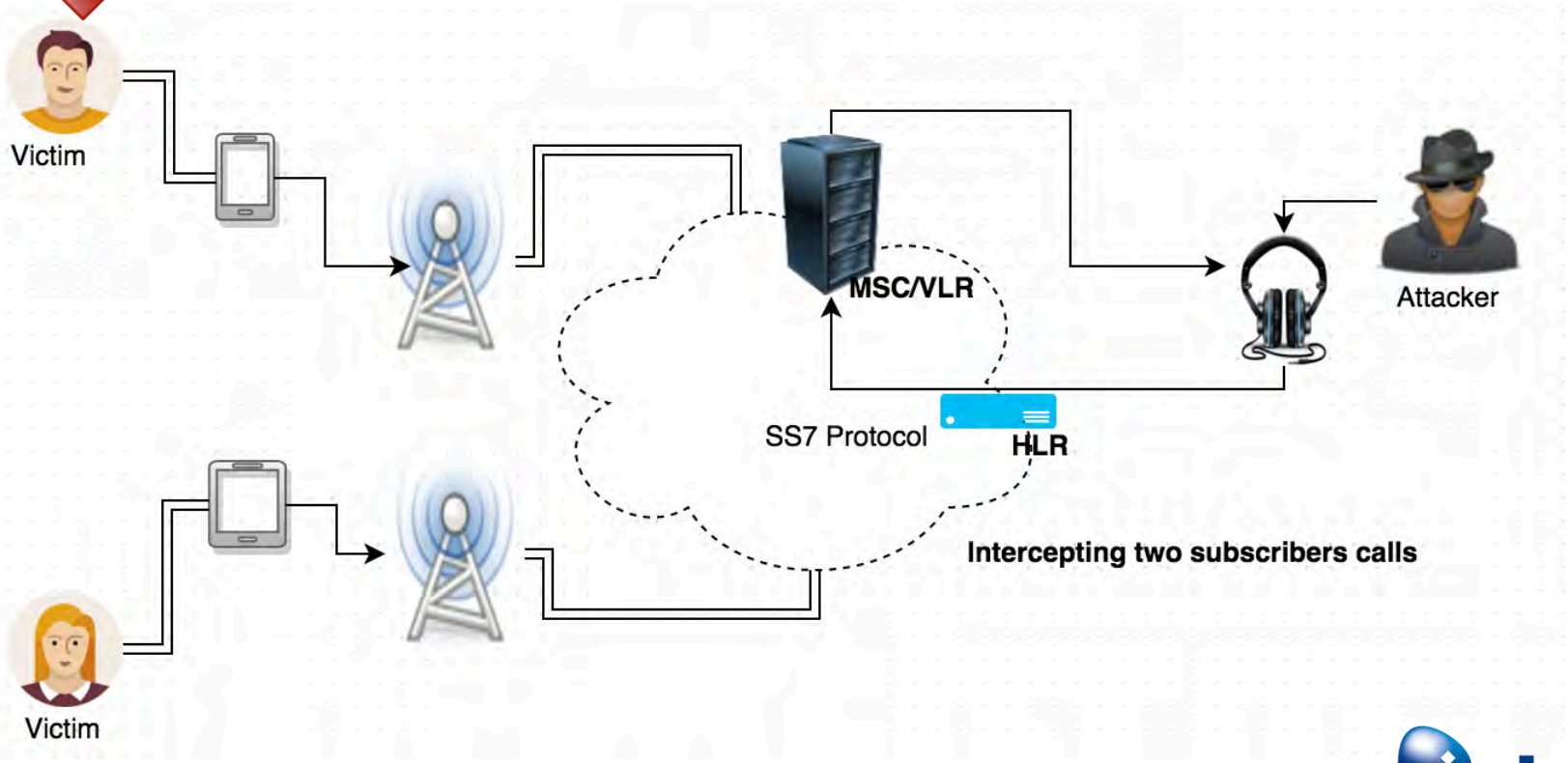


kpn

SS7 Practical Attack Scenarios

1

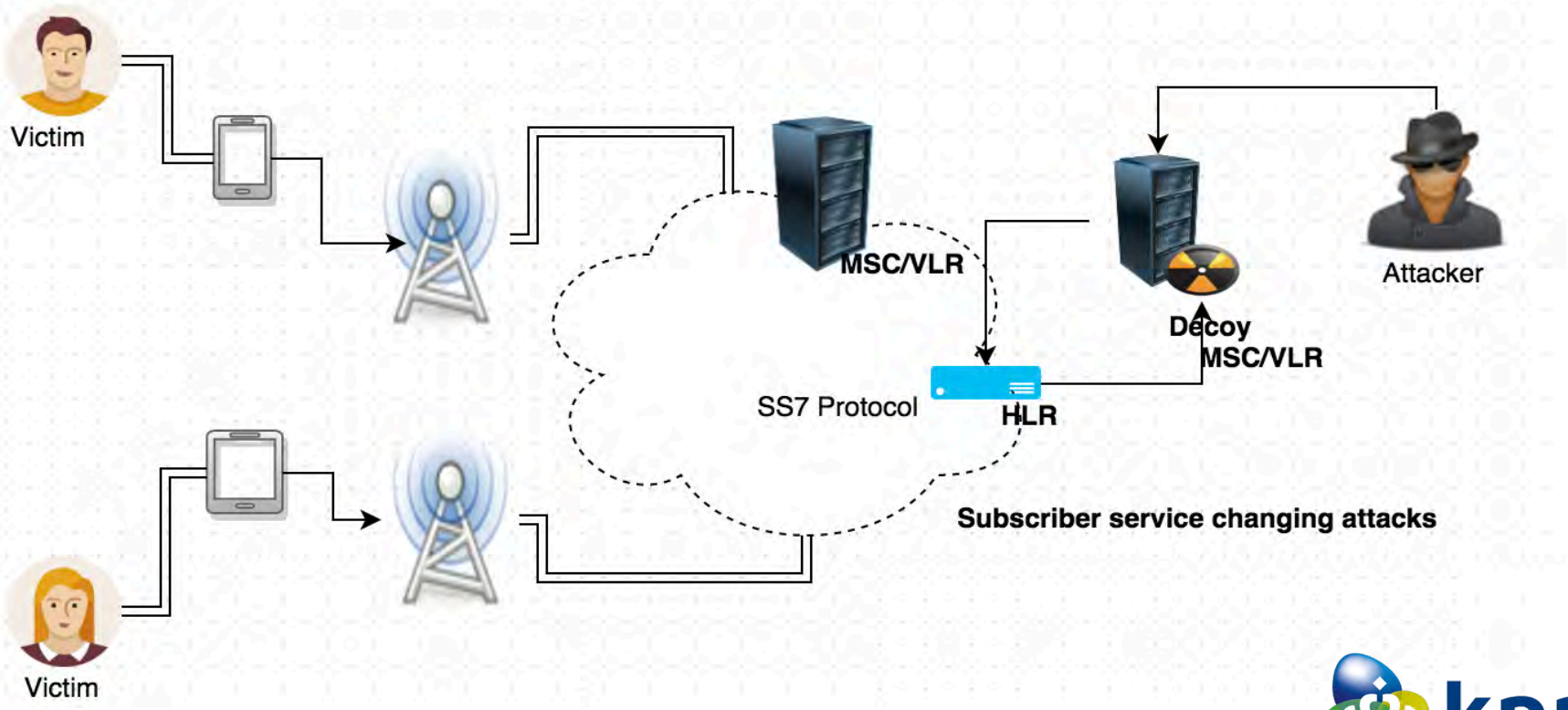
- Intercepting subscribers calls



SS7 Practical Attack Scenarios

2

- Subscriber service change attacks





SS7 Practical Attack Scenarios

31

3

- Interception of SMS messages

4

- Interception of outgoing calls

5

- Redirection of incoming or outgoing calls

6

- Making changes in user bills or balance

kpn



SS7 Practical Attack Scenarios

32

7

- Unblocking stolen mobile devices

Unblocking stolen mobile devices using SS7-MAP vulnerabilities

Exploiting the relationship between IMEI and IMSI for EIR access

Siddharth Prakash Rao[1, 2], Silke Holtmanns[1], Ian Oliver[1], Tuomas Aura[2]
[1]Nokia Networks
Espoo, Finland
{sid.rao, silke.holtmanns, ian.oliver}@nokia.com
[2]Department of Computer Science and Engineering
Aalto University, School of Science
Espoo, Finland
{siddharth.rao, tuomas.aura}@aalto.fi

***Abstract*— The increase in usage of mobile phones and the relative increase in the number of mobile phone thefts have imposed an overhead on securely retrieving the stolen or missing devices. While the mobile security researchers try to figure out various mechanisms to track such devices, attackers on the other hand are trying to exploit weaknesses in the mobile network system to dissipate into the dark side with stolen devices. In this paper we present how the SS7- MAP protocol can be misused to help an attacker to unblock the device from the stolen list and use it normally.**

***Keywords*— SS7, MAP, EIR, Stolen Mobiles, IMEI**

over stolen/missing devices and the private information on them. There exists a growing black market [3] where such devices are sold with least possibility of tracing them.

Signaling System No.7 (SS7) is a mobile backend protocol used for interconnectivity between mobile operator networks which enables roaming and cellular services across operator domains. This protocol is mainly used for communication between the network elements and between networks. Recent successful attacks on SS7 as per [4][5][6][7] and [8] haven proven that an attacker with privilege over the core mobile network can take control over users personal information such as billing data and Short Message Service (SMS) messages

IEEE August 2015, Nokia Researchers Espoo, Finland.

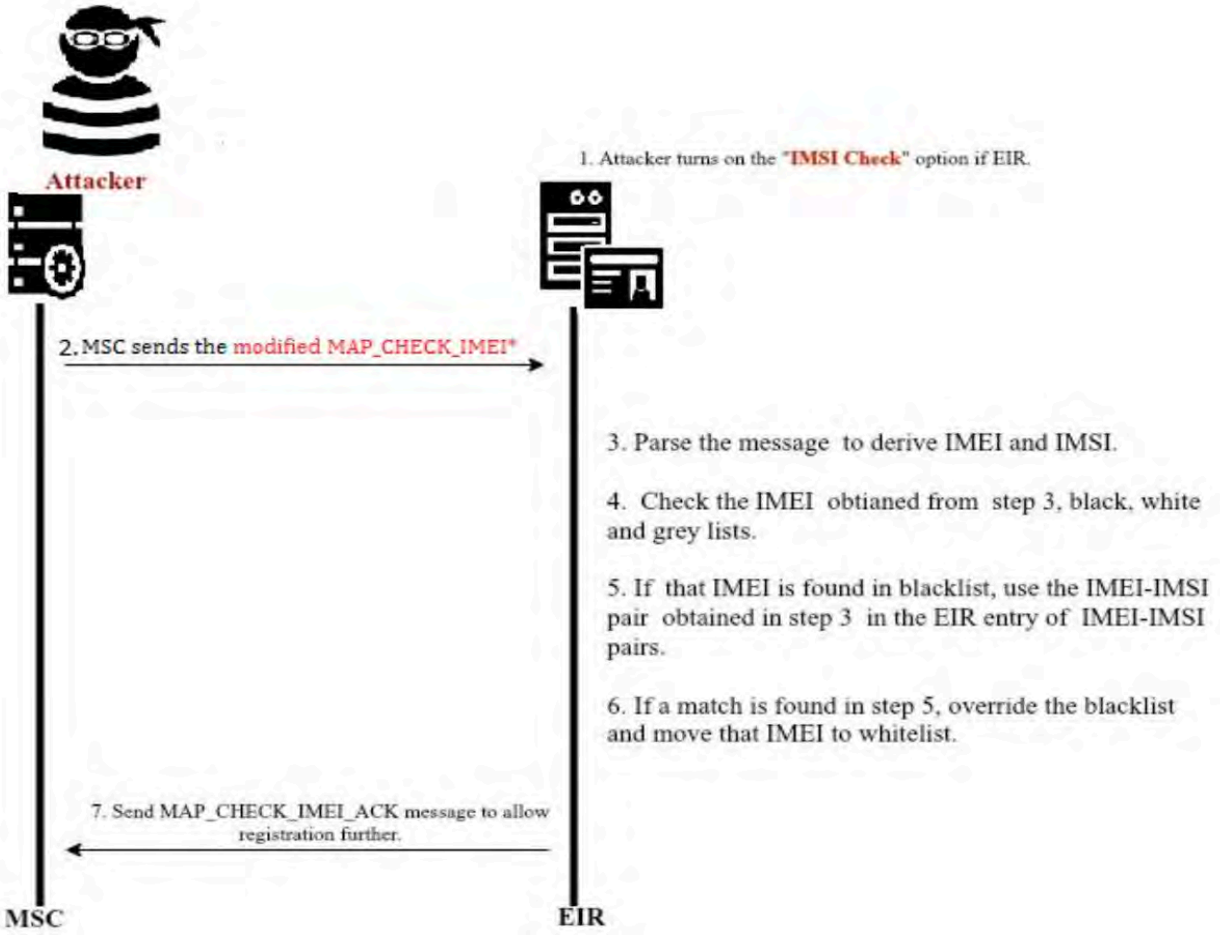


SS7 Practical Attack Scenarios

33

7

- Unblocking stolen mobile devices



IEEE August 2015, Nokia Researchers Espoo, Finland.

Hacking Team after SS7 Hacks



34

On 5 Sep, 2013, at 3:10 PM, Alberto Ornaghi <a.ornaghi@hackingteam.com> wrote:

I think that any telco in the world could access SS7 signals to know where a phone number is located. if the have to call that number from a country, let's say UK, the UK's provider must know where the phone is attached to the global network since if the called one is in UK the call is not routed abroad and routed directly internally.
sounds plausible?
On Sep 5, 2013, at 09:03 , serge <s.woon@hackingteam.com> wrote:

To narrow down the possible whistleblower, I think the person has to be working in the telecom provider we subscribed to ... TIM?Location query from SS7 signalling can only indicate the location of the mobile phone at the time of query (we can verify this with NICE :-)). The records are clearly a database query as it is very systematic; twice per day at 1100h and 2300h. As the records started from one or two years back, either Mus and Marco.B have a common secret admirer (which I do not think so) or the person just query the records from the telco existing database knowing that their names are mentioned in controversial articles. We can compliant to TIM (I am not sure whether they will do anything about it) or just switch to another telco and monitor the situation.

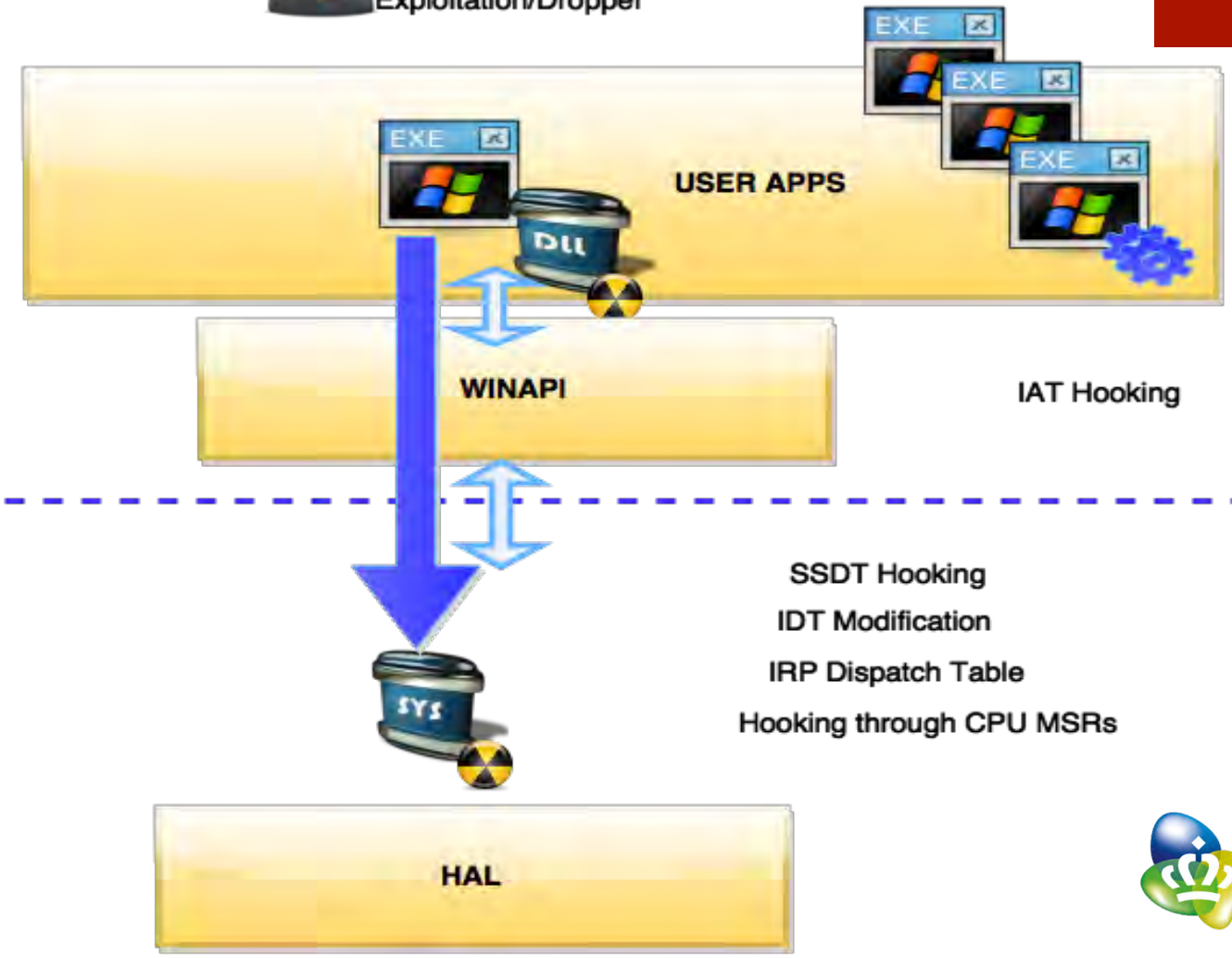
Source: https://wikileaks.org/hackingteam/emails/emailid/343623



Rootkit Techniques



Exploitation/Dropper



kpn



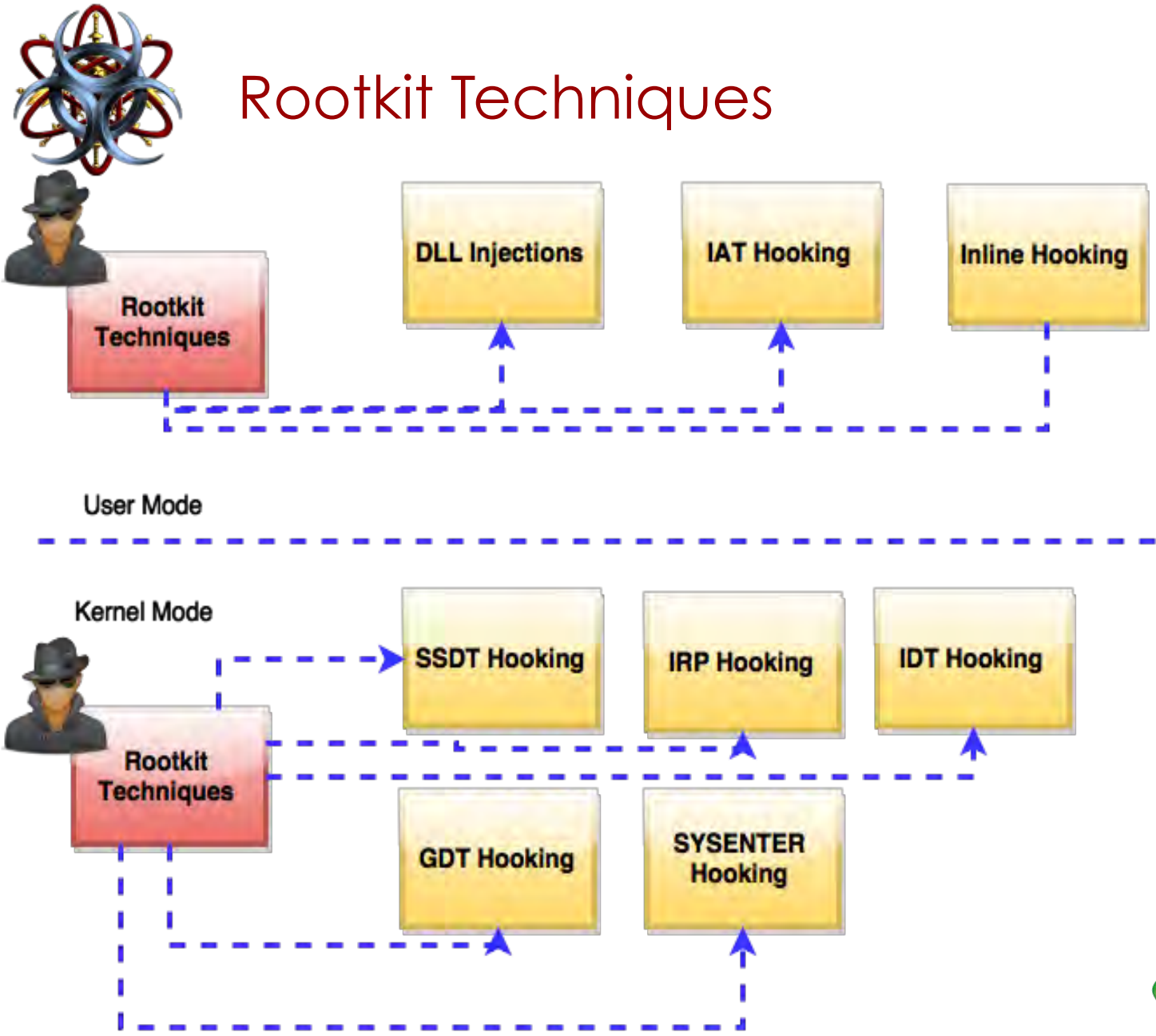
Hardware/Software Interception: Captain Hook Style Hacking



Captain Hook Style Hacking: *Intercepts every function, keeps a copy of the content for herself, and then let the function continue as it was supposed to …*

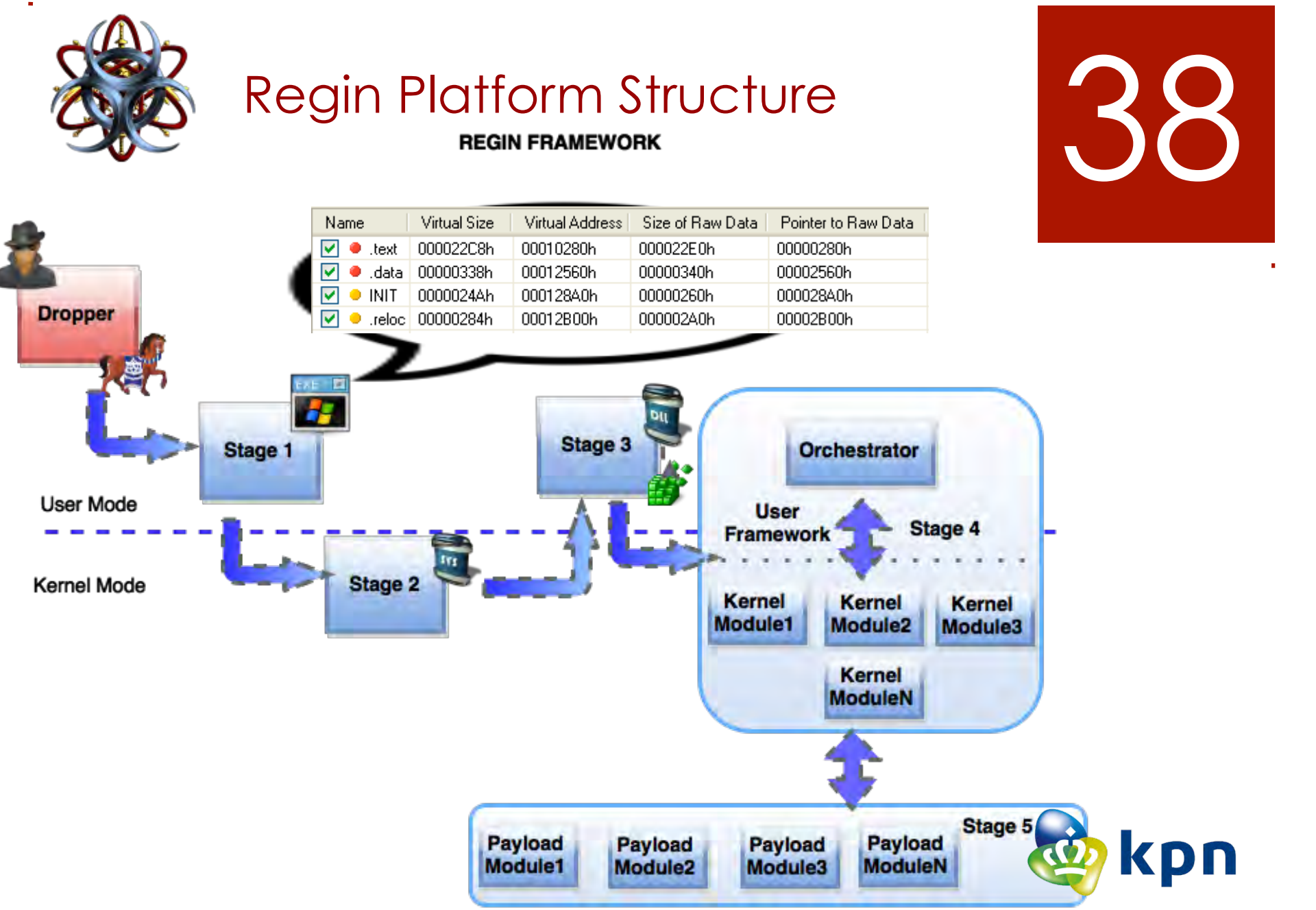
kpn

Rootkit Techniques



kpn

Regin Platform Structure



Regin Platform Analysis

- Challenges, Hurdles & Difficulties:



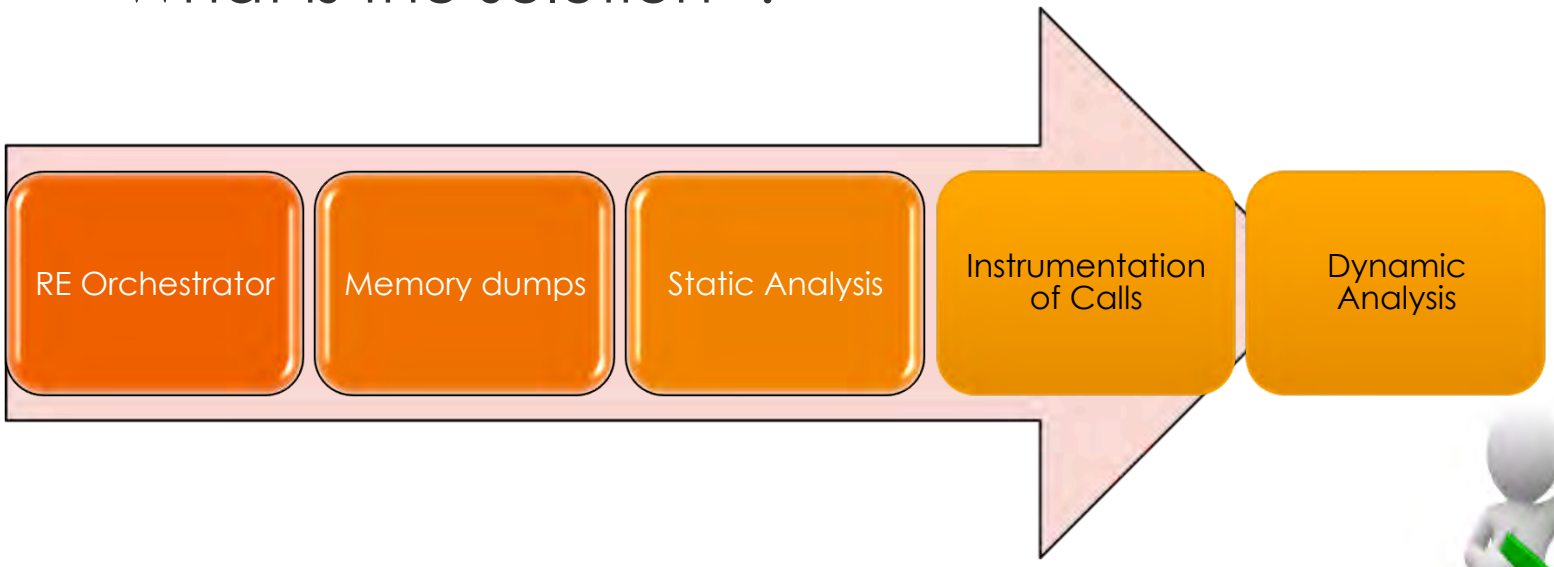
- No one had the dropper when started analysis
- Multi stage and encrypted framework structure
- Modules are invoked via SOA structure by the framework
- Malware data are stored inside the VFS
- Researched GSM Networks had no indication of compromise ☺

kpn



Regin Platform Analysis

- What is the solution ?

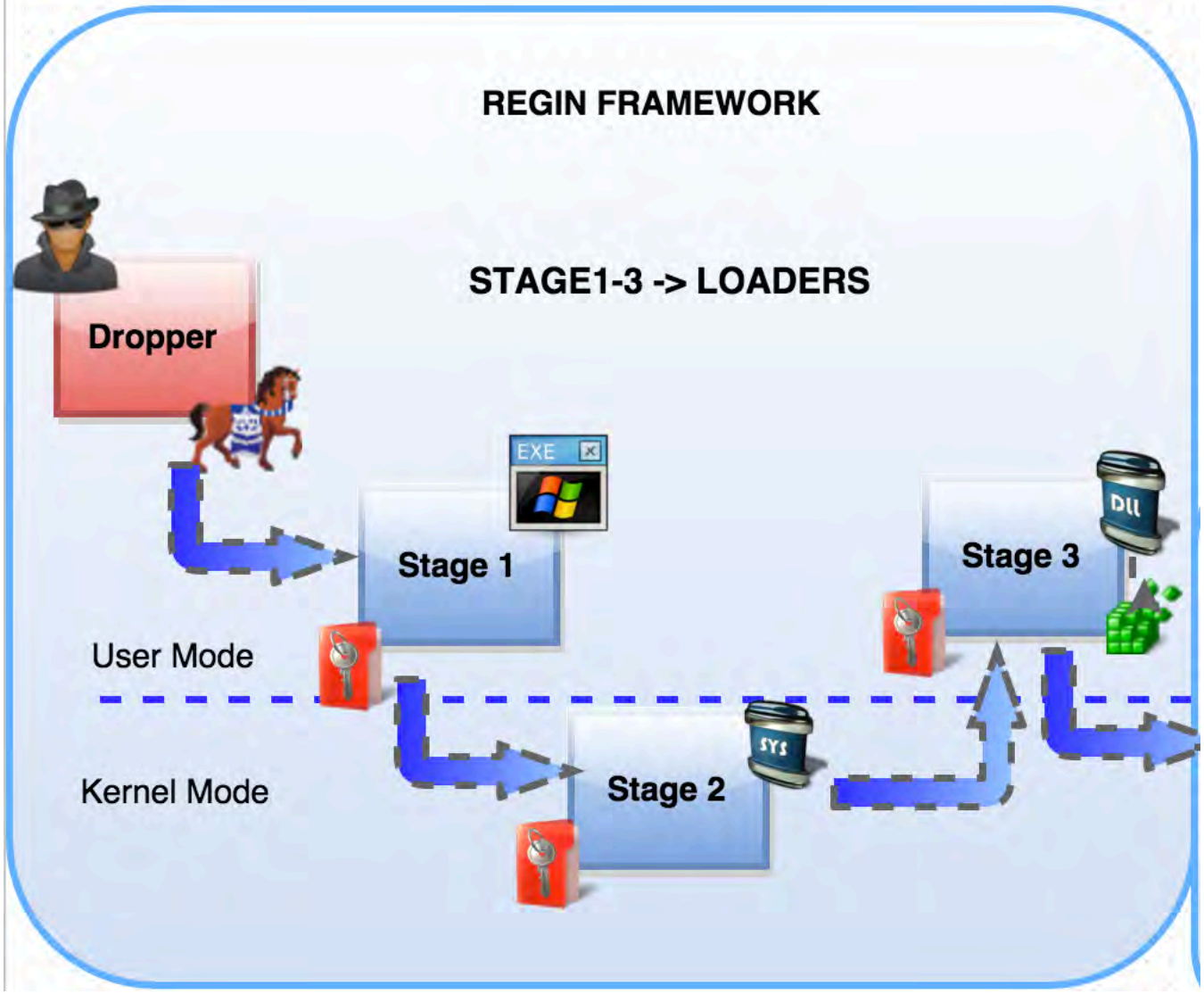




Check similar work & the write up:
http://artemonsecurity.com/regin_analysis.pdf

kpn

Regin Platform Stages

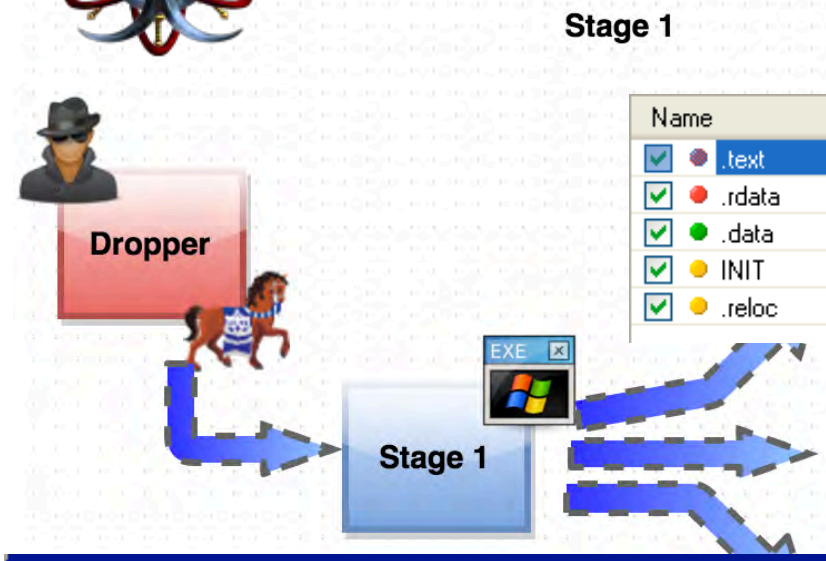


kpn

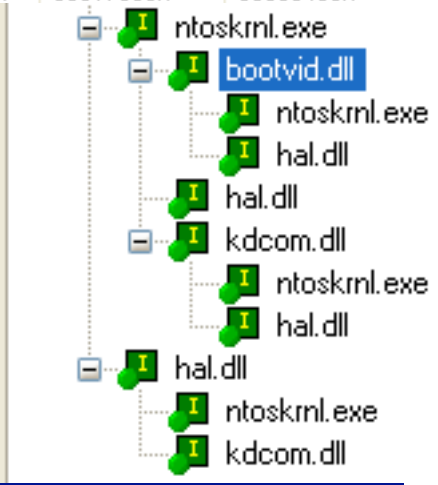


Regin Platform – Stage 1

42



Name	Virtual Size	Virtual Address	Size of Raw Data	Pointer to Raw Data
.text	00002578h	00011000h	00002600h	00000400h
.rdata	00000104h	00014000h	00000200h	00002A00h
.data	0000033Ch	00015000h	00000400h	00002C00h
INIT	000002E0h	00016000h	00000400h	00003000h
.reloc	00000396h	00017000h	00000400h	00003400h



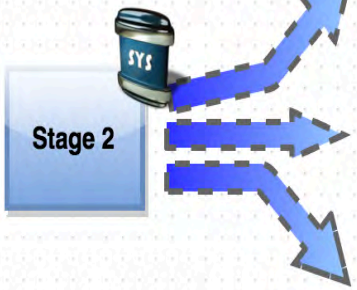

```
        extrn memcpy
        extrn RtlTimeToTimeFields
        extrn ExSystemTimeToLocalTime
        extrn KeQuerySystemTime
        extrn ExFreePool
        extrn NtBuildNumber
        extrn ZwQuerySystemInformation
        extrn ZwCreateFile
        extrn memset
        extrn ZwQueryInformationFile
        extrn ExAllocatePoolWithTag
```

Unprocessed data | Strings | View 1 | View 2 | View 3 | View 4

```
00011088                        jmp_ntoskrnl.exe!_except_handler3:
00011088  FF2578400100                  jmp     [ntoskrnl.exe!_except_handler3]
0001108E                        SUB_L0001108E:
0001108E  8BFF                          mov     edi,edi
00011090  55                            push    ebp
00011091  8BEC                          mov     ebp,esp
00011093  83EC20                        sub     esp,00000020h
00011096  8B450C                        mov     eax,[ebp+0Ch]
00011099  C645FF00                      mov     byte ptr [ebp-01h],00h
```



Regin Platform – Stage 2




```
ObjectAttributes= OBJECT_ATTRIBUTES ptr -34h
DestinationString= UNICODE_STRING ptr -1Ch
ValueName= UNICODE_STRING ptr -14h
Length= dword ptr -0Ch
KeyHandle= dword ptr -8
ResultLength= dword ptr -4

mov     edi, edi
push    ebp
mov     ebp, esp
sub     esp, 34h
push    esi
mov     esi, ds:RtlInitUnicodeString
push    edi
push    offset SourceString ; "\\REGISTRY\\Machine\\System\\CurrentCon"
lea     eax, [ebp+DestinationString]
xor     edi, edi
push    eax             ; DestinationString
mov     [ebp+ResultLength], edi
call    esi ; RtlInitUnicodeString
push    offset word_163E6 ; SourceString
lea     eax, [ebp+ValueName]
push    eax             ; DestinationString
call    esi ; RtlInitUnicodeString
lea     eax, [ebp+DestinationString]
mov     [ebp+ObjectAttributes.ObjectName], eax
lea     eax, [ebp+ObjectAttributes]
push    eax             ; ObjectAttributes
push    0F003Fh         ; DesiredAccess
```

```
120         RegistryPatha = RtlWriteRegistryValue(
121                             4u,
122                             SourceString.Buffer,
123                             Destination.Buffer,
124                             1u,
125                             ValueData,
126                             (unsigned __int16)v18 + 2);
127         if ( RegistryPatha < 0 )
128           break;
129       }
130       Destination.Length -= DestinationString.Length;
131       ObjectName.Length -= DestinationString.Length;
```

kpn

Regin Platform – Stage 2

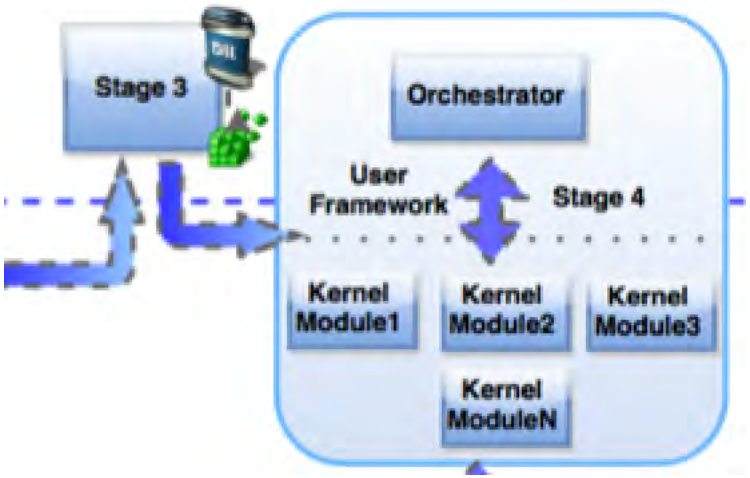

```
12  ResultLength = 0;
13  RtlInitUnicodeString(
14    &DestinationString,
15    L"\\REGISTRY\\Machine\\System\\CurrentControlSet\\Control\\Session Manager\\Memory Management");
16  RtlInitUnicodeString(&ValueName, &word_163E6);
17  ObjectAttributes.ObjectName = &DestinationString;
18  ObjectAttributes.Length = 24;
19  ObjectAttributes.RootDirectory = 0;
20  ObjectAttributes.Attributes = 64;
21  ObjectAttributes.SecurityDescriptor = 0;
22  ObjectAttributes.SecurityQualityOfService = 0;
23  result = ZwOpenKey(&KeyHandle, 0xF003Fu, &ObjectAttributes);
24  if ( !result )
25  {
26    if ( ZwQueryValueKey(KeyHandle, &ValueName, KeyValueFullInformation, 0, 0, &ResultLength) == -1073741789 )
27    {
28      Length = ResultLength;
29      v1 = sub_12892(ResultLength, PagedPool);
30      if ( v1 )
31      {
32        if ( !ZwQueryValueKey(KeyHandle, &ValueName, KeyValueFullInformation, (PVOID)v1, Length, &ResultLength)
33          && *(_DWORD *)(v1 + *(_DWORD *)(v1 + 8)) == 1 )
34          byte_19B79 = 1;
35        sub_12952((PVOID)v1, Length);
```

```
85  memset(v5, 0, ObjectName.MaximumLength);
86  RtlAppendUnicodeToString(&ObjectName, L"\\Device\\");
87  RtlAppendUnicodeToString(&ObjectName, v16.Buffer);
88  v14 = ExAllocatePoolWithTag(0, 4 * v12, 0x206B6444u);
89  if ( v14 )
90  {
91    v22 = 0;
92    for ( Value = 0; Value < v12; ++Value )
93    {
94      RegistryPatha = RtlIntegerToUnicodeString(Value, 0xAu, &DestinationString);
```

kpn



Regin Platform – Stage 3 & 4



ExAllocatePoolWithTag routine

The **ExAllocatePoolWithTag** routine allocates pool memory of the specified type and returns a pointer to the allocated block.

Syntax

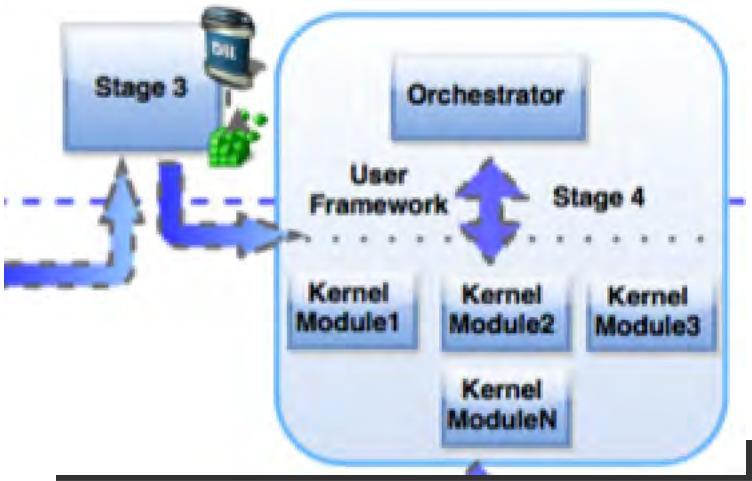
C++

```
PVOID ExAllocatePoolWithTag(
  _In_ POOL_TYPE PoolType,
  _In_ SIZE_T    NumberOfBytes,
  _In_ ULONG     Tag
);
```

```
00029232 aNtoskrnl_exe   db 'ntoskrnl.exe',0
0002923F                 align 10h
00029240 unk_29240       db  3Ah ; :             ; DATA XREF: seg000:000291C8↑o
00029241                 align 2
00029242 aExallocatepool db 'ExAllocatePoolWithTag',0
00029258 unk_29258       db  47h ; G             ; DATA XREF: seg000:000291C4↑o
00029259                 align 2
```

kpn

Regin Platform – Stage 3 & 4



C++

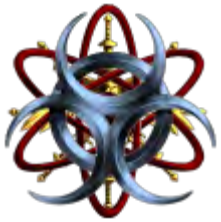
```
VOID KeStackAttachProcess(
   _Inout_ PRKPROCESS    Process,
   _Out_   PRKAPC_STATE ApcState
);
```

Hex View-1 | Structures | Enums | Imports

```
:000294E1                   db    3
:000294E2 aPsterminatesys   db 'PsTerminateSystemThread',0
:000294FA                   db  22h ; "
:000294FB                   db    2
:000294FC aKeunstackdetac   db 'KeUnstackDetachProcess',0
:00029513                   align 4
:00029514                   db 0F0h ; ≡
:00029515                   db    2
:00029516                   db  50h ; P
:00029517 aScreatesystemt   db 'sCreateSystemThread',0
:0002952B                   align 4
:0002952C                   db  1Eh
:0002952D                   db 2, 4Bh, 65h
:00029530 aStackattachpro   db 'StackAttachProcess',0
:00029543                   align 4
:00029544                   db  60h ; `
:00029545                   db 4, 5Ah, 77h
:00029548                   db  4Fh ; O
:00029549 aPenprocess       db 'penProcess',0
:00029554                   db 0F3h ; ≤
:00029555                   db    2
```

00029548: seg000:00029548

kpn



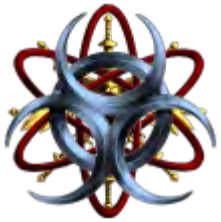
Regin Platform – Stage 3 & 4 – How to Weaponize it ?

47

1. • Register a call-back function to a process
2. • Log the PID of the target process
3. • Obtain PEB via **ZwQueryInformation()** for base adresses of the modules
4. • Obtain the EP via **PsLookupProcesByProcess**()
5. • Get inside to the process context via **KeStackAttachProcess()** referenced by EP
6. • Read PEB and other data in process context



kpn

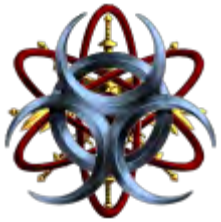


Regin Platform – Stage 3 & 4 – How to Weaponize it ?


```
InitializeObjectAttributes(&ObjectAttributes, NULL, OBJ_KERNEL_HANDLE, NULL,
    NULL);

clientID.UniqueProcess = hProcessId;
clientID.UniqueThread = NULL;


__try{


    status = ZwOpenProcess(&hProcessHandle, DesiredAccess, &ObjectAttributes,
        &clientID);

    status = ZwQueryprocess(hProcessHandle, ProcessBasicInformation,
        (PVOID)&BasicInfoReal, sizeof(PROCESS_BASIC_INFORMATION), &SizeReturned);

    status = PsLookupProcessByProcessId(hProcessId, &ep);

    __asm{
        mov eax, ep
            mov eax, [eax]
            mov myPwnProcess, eax
    }

    KeStackAttachProcess(&myPwnProces, ka_state);
```



kpn



Uruborus < Regin < Duqu2

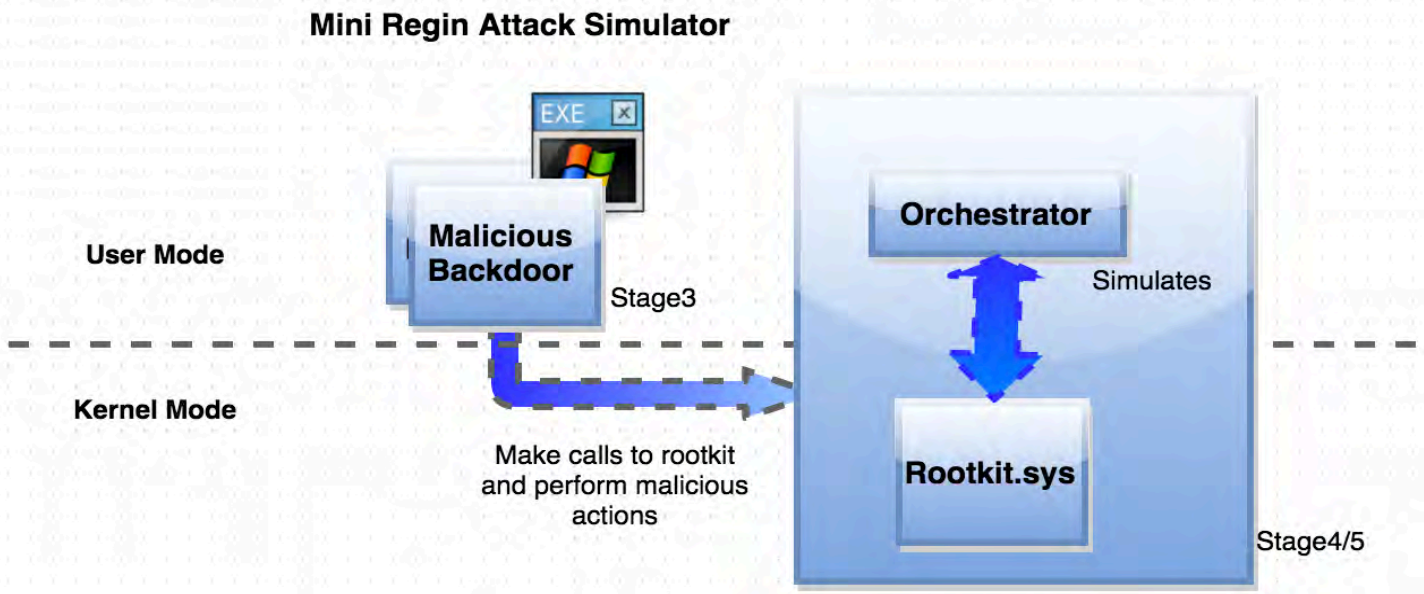


Uruborus	Regin	Duqu2
Encrypted VFS	Encrypted VFS	Encrypted VFS #2
PatchGuard Bypass	Fake Certificate	Stolen Certificate
Multiple Hooks	Orchestrator SOA	Orchestrator SOA
AES	RC5	Camellia 256, AES, XXTEA
Backdoor/Keylogger Mod	Advanced Network/ File Mods	More Advanced Network/File/USB Mods

kpn



Regin Attack Simulation



Mini Regin Attack Simulator
Covert Channel Data Exfiltration
Run as a thread of legitimate app's address space
Orchestrator simulator and partial SOA
File system, registry and network calls hooking
Backdoor/Keylogger Mod

kpn

Demo



kpn

Questions ?



Thank you very much for your attention







kpn



References

- http://denmasbroto.com/article-5-gprs-network-architecture.html
- http://docstore.mik.ua/univercd/cc/td/doc/product/wireless/moblwrls/cmx/mmg_sg/cmxgsm.htm
- http://4g-lte-world.blogspot.nl/2013/03/gprs-tunneling-protocol-gtp-in-lte.html
- http://labs.p1sec.com/2013/04/04/ss7-traffic-analysis-with-wireshark/
- http://www.gl.com/ss7_network.html
- http://www.slideshare.net/mhaviv/ss7-introduction-li-in
- http://www.gl.com/ss7.html

kpn