Integrated content security technology and why we need it

Jimmy Shah
Antivirus Researcher

Contents

Who we are

Mobile Malware

Mobile Vulnerabilities

Why we need integrated client side content scanning

Who we are

Who we are

McAfee

- Mobile Antivirus Researchers
- My team and I specialize in mobile malware and threat analysis on existing(J2ME, SymbOS,WM, iPhone OS, Android) and upcoming mobile platforms.
- We work with a number of large mobile network operators.

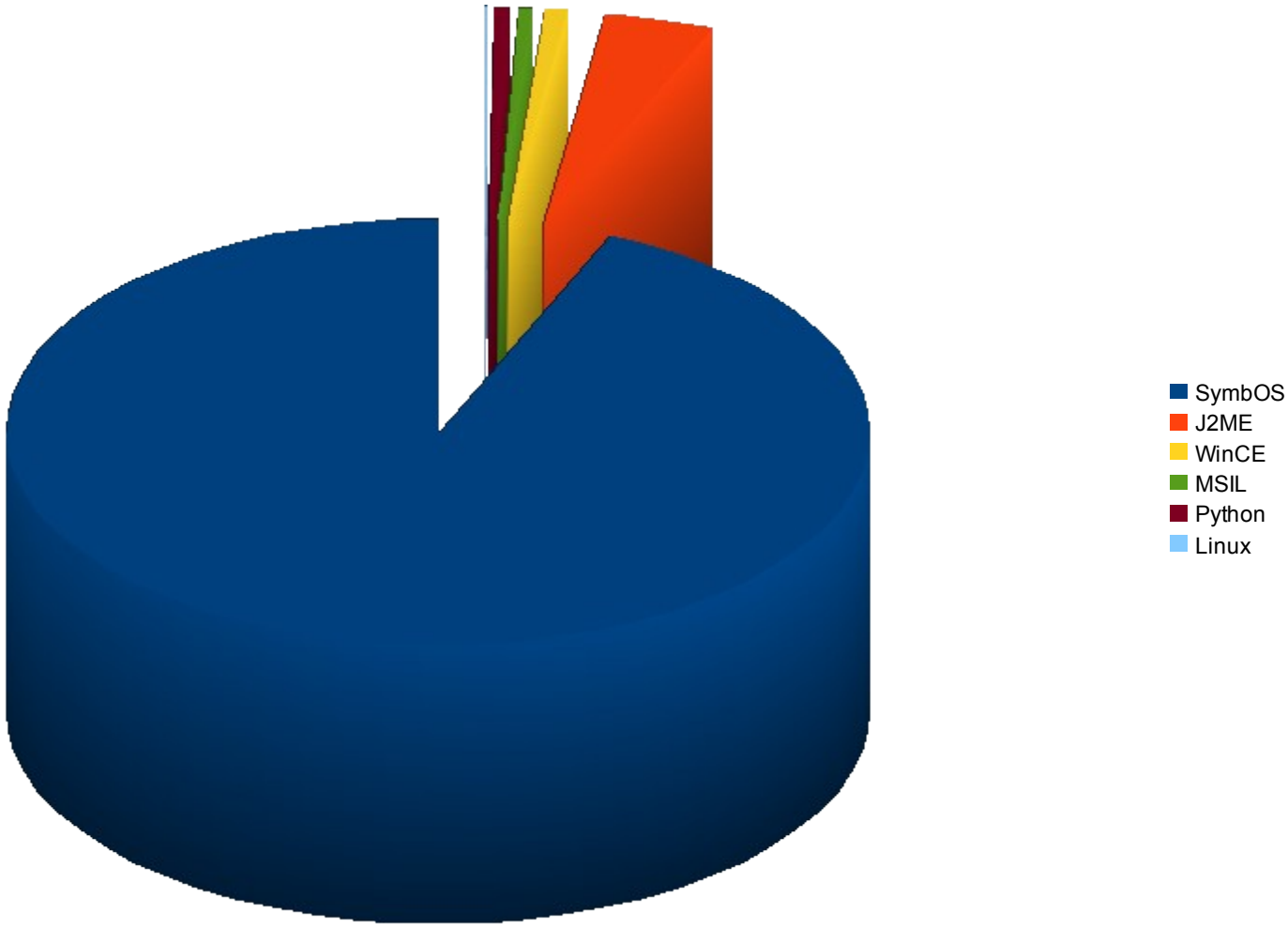
Mobile Malware

Types of Mobile Phones

McAfee

- Smartphones
  - Full OS
    - Symbian, Windows CE, Blackberry, iPhone, Android
  - Installable software
  - Tons of bells and whistles
- Featurephones
  - “Why can't I have a phone that is just a phone?”

Mobile Malware by Platform



Mobile Malware Lifecycle

McAfee

- R & D
  - Develop Proof of Concepts(PoCs)
  - Obfuscation, Packing, other detection evasion
- Reuse
  - Copy-cats, Script kiddies, lesser skilled malware authors
- Profit taking
  - Premium Rate fraud (Ringtone/wallpaper subscriptions, Adult sites, etc.) (J2ME/Redbrowser.A,J2ME/Wesber.A)
  - Virtual currency fraud(QQ coins) (SymbOS/Multidropper.CR)
  - Money transfer fraud (Python/Refloc.A)

Installed malware countermeasures

McAfee

- Code signing
  - Individual programs/installers are signed
    - No certificate, no install
    - Modifications of executables prevents them from being run.
      - Goodbye file infectors
    - Exploitable APIs are restricted
  - Firmware updates are signed
  - Developers sign their work

Mobile Vulnerabilities

DCM 110 dialer

- Exploited cHTML in order to cause a DoS
  - Clicking the link causes the phone to dial the 110 emergency number
  - Technical measures used to neutralize attack

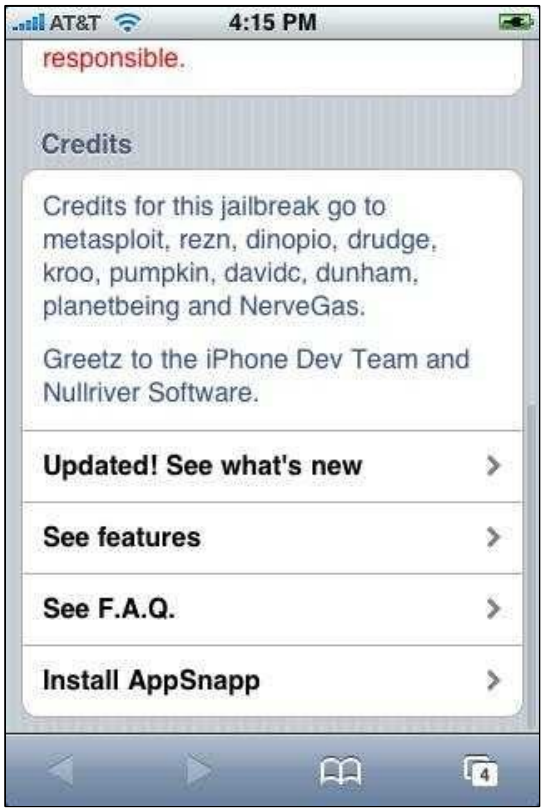

```
<HTML>
<BODY>
<P>Brave Heart Room
<P><a href="tel:911">
Click here if you are
not a chicken</a>
</BODY>
</HTML>
```

libTIFF vulnerability

McAfee

- Playstation Portable(PSP) hacker Niacin discovered it could crash the PSP in 2006
  - Exploited to install bootloader and run “custom” software
- Same hacker, same vuln on iPhone in 2007
  - Exploited to allow jailbreaking
- AppSnapp
  - Visit http://www.jailbreakme.com with a 1.2 iPhone for one-click jailbreak.
  - The site uses the libTIFF exploit to add an installer application to the iPhone
  - The vulnerability is patched at the end of the process

libTIFF vulnerability, cont.



Multiple Nokia J2ME vulnerabilities

McAfee

- Discovered by Adam Gowdiak
  - As member of Last Stage of Delirium(LSD) security group, he discovered Java vulnerabilities in PC browsers in 2002.
- Vulnerabilities found in
  - KVM(K Virtual Machine)
  - Nokia 6310i
- Reverse engineered Nokia OS and filesystem access(overwrite flash memory)
- PoC MIDlets could make calls, send SMS, access the Internet and read /write files on the phone without user permission.

Screenshots of the Java debug agent from Gowdiak's 2004 presentation:

Multiple Nokia J2ME vulnerabilities, cont.

McAfee

- Adam Gowdiak extended his original research, targeting newer Nokia S40, 3rd edition phones
  - Charged €20,000 for access to the research report and PoC malicious samples
  - Sun and Nokia have confirmed the existence of the vulnerabilities
- Malicious MIDlets were created that were capable of:
  - gaining additional privileges for a malicious MIDlet, even manufacturer or mobile carrier level
  - running a malicious MIDlet when the phone is first turned on
  - accessing files and the SIM card
  - sending SMS/MMS
  - reading contacts and making phone calls
  - eavesdropping using the camera and microphone
  - installing a backdoor program that allows all of the above to be performed remotely

First iPhone Exploits

McAfee

- Vulnerability discovered by Charlie Miller of Independent Security Evaluators(ISE)
  - bug in the outdated version of the PCRE library installed on the iPhone
  - discovered by fuzzing
- Two exploits created
  - SMS messages, contacts and other info are sent to the attacker
  - Vibrating the iPhone

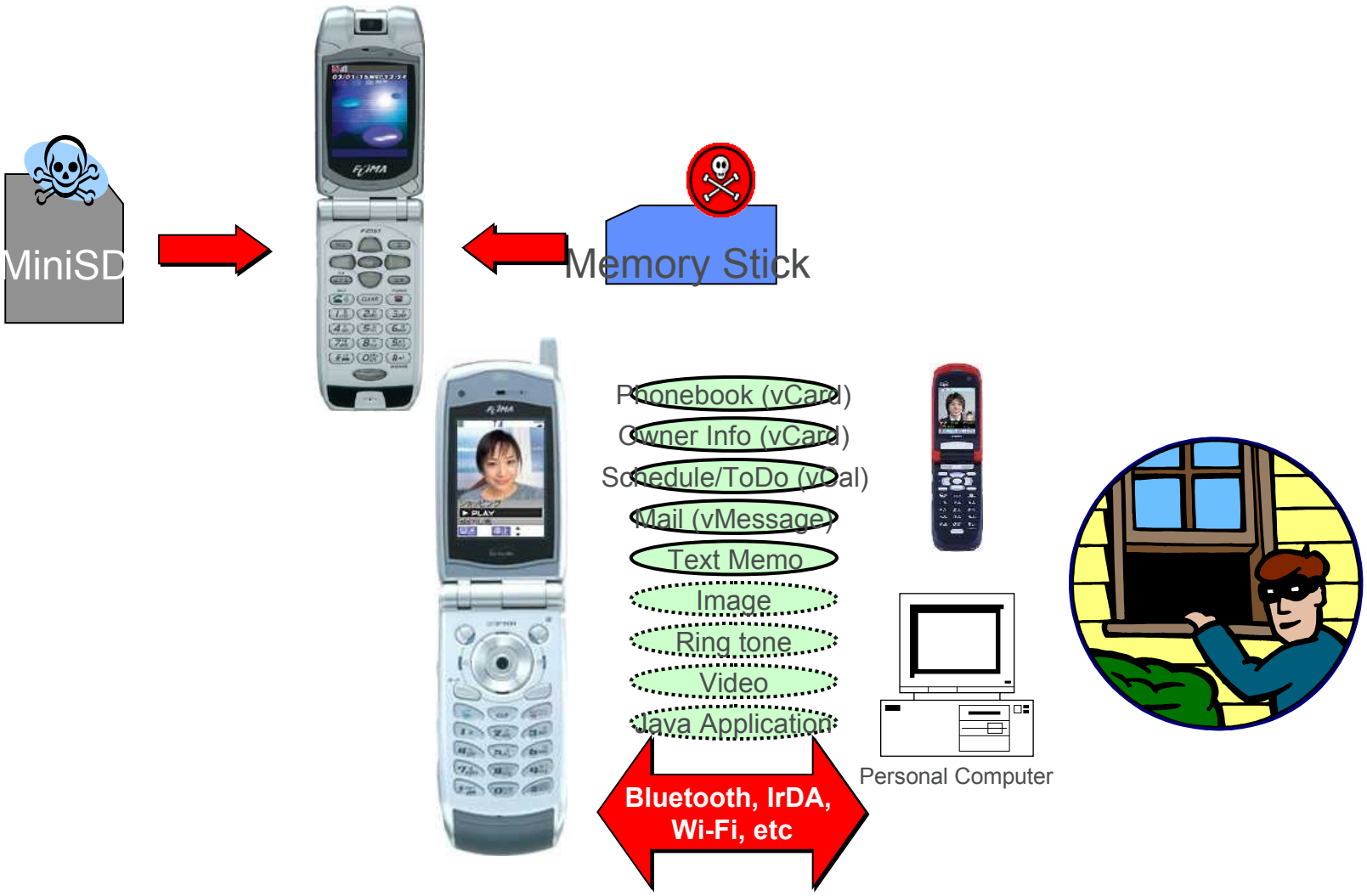
T-Mobile G1 Browser exploit

McAfee

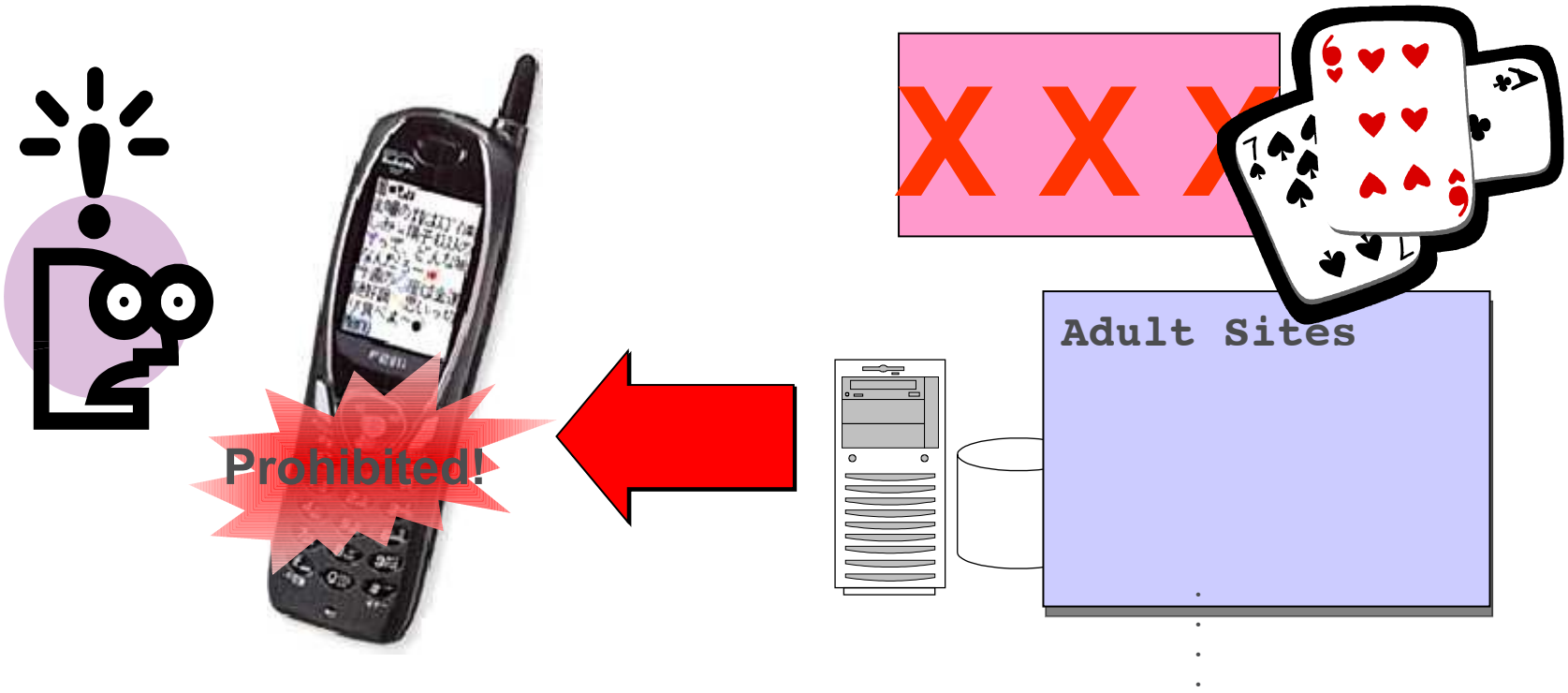
- Charlie Miller discovered a vulnerability in the Android Browser pre-release of the G1
- Specially crafted Javascript w/ regex can trigger a vulnerability in the G1's Webkit based browser
- A successful exploit can allow an attacker to access browser cookies and log keystrokes

Why we need integrated client side content scanning

Content received from local environment

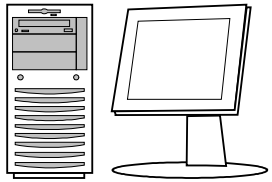


Content that connects to premium sites



Content that causes misbehaviors on mobiles

Malformed Image data

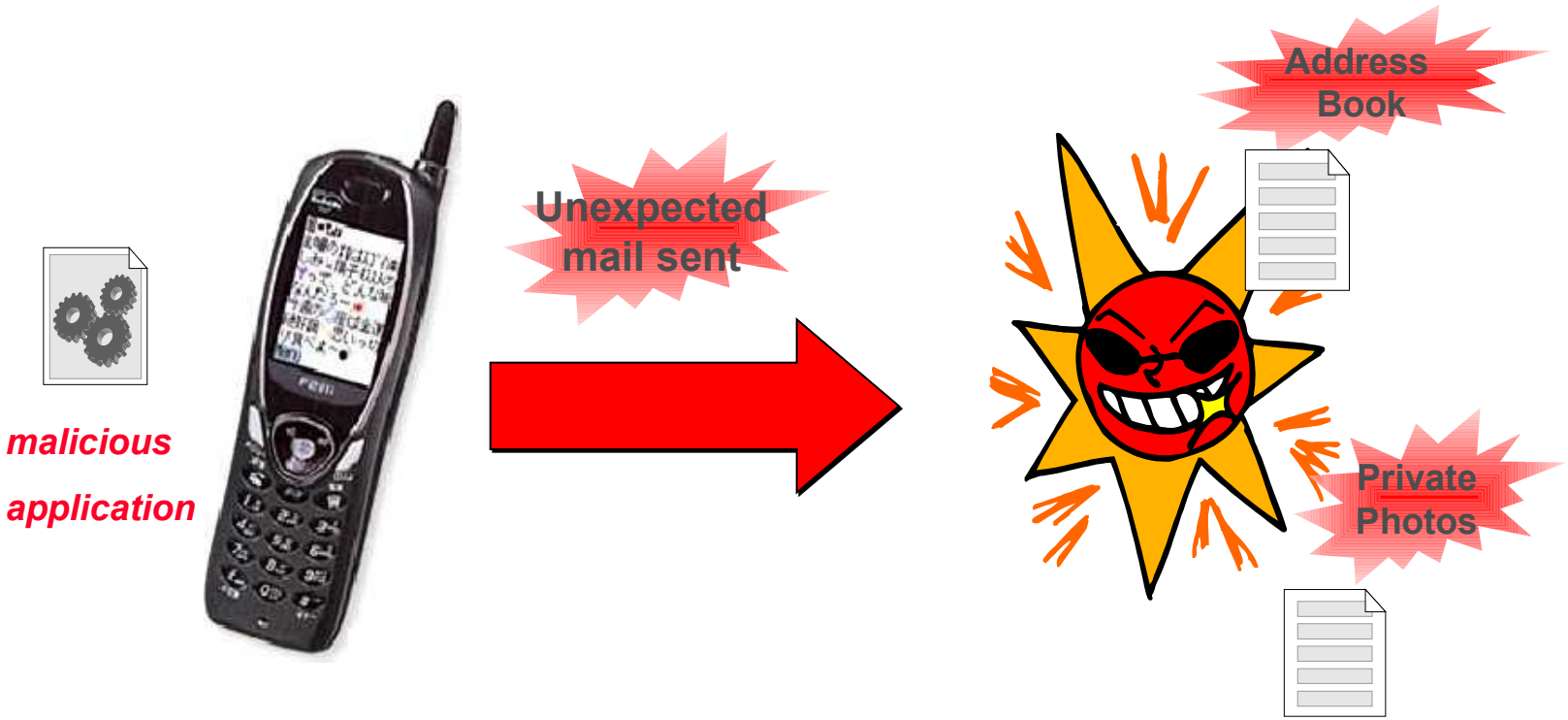


address	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
0C00:0000	FF	D8	FF	E0	00	10	4A	46	49	46	00	01	02	00	00	64
0C00:0010	00	64	00	00	FF	EC	00	11	44	75	63	6B	79	00	01	00
0C00:0020	04	00	00	00	3C	00	00	FF	FE	00	26	41	64	6F	62	65
0C00:0030	00	64	C0	00	00	C0	01	03	00	15	04	03	06	0A	00	00
0C00:0040	00															
0C00:0050	DB															
0C00:0060	05															
0C00:0070	00															
0C00:0080	03															
0C00:0090	FF															
0C00:00A0	5A															
0C00:00B0	FF															
0C00:00C0	FF															
0C00:00D0	FF	FF	FF	FF	FF	FF	C2	00	11	08	00	D4	00	DF	03	01
0C00:00E0	01	C0	02	11	01	C3	11	01	FF	C4	00	EA	00	00	02	03
0C00:00F0	01	C1	01	00	00	C0	00	00	00	00	00	00	00	00	04	02
0C00:0100	03	C5	01	06	07	C1	01	01	01	01	01	01	00	00	00	00

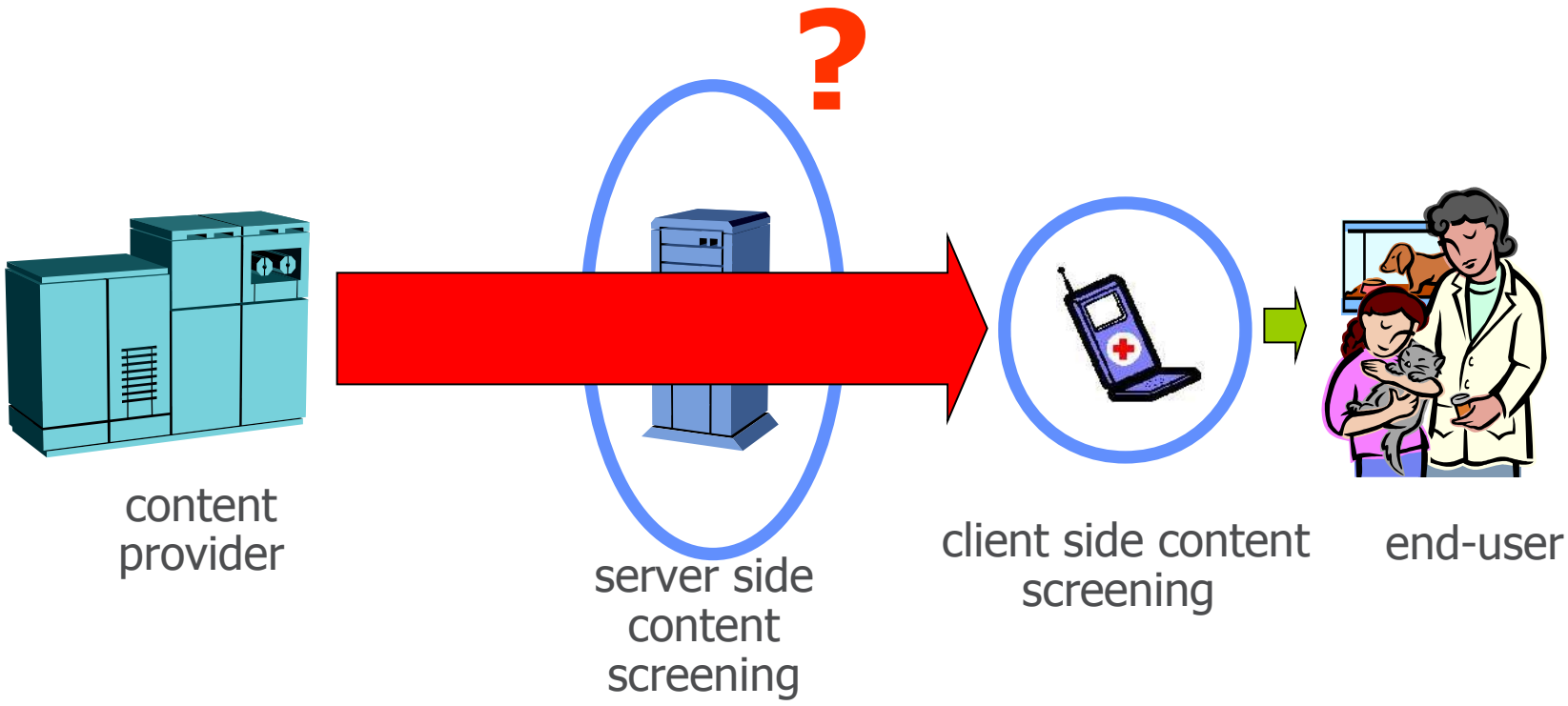
Length of data in image header is longer than actual value.

Content accessed while roaming

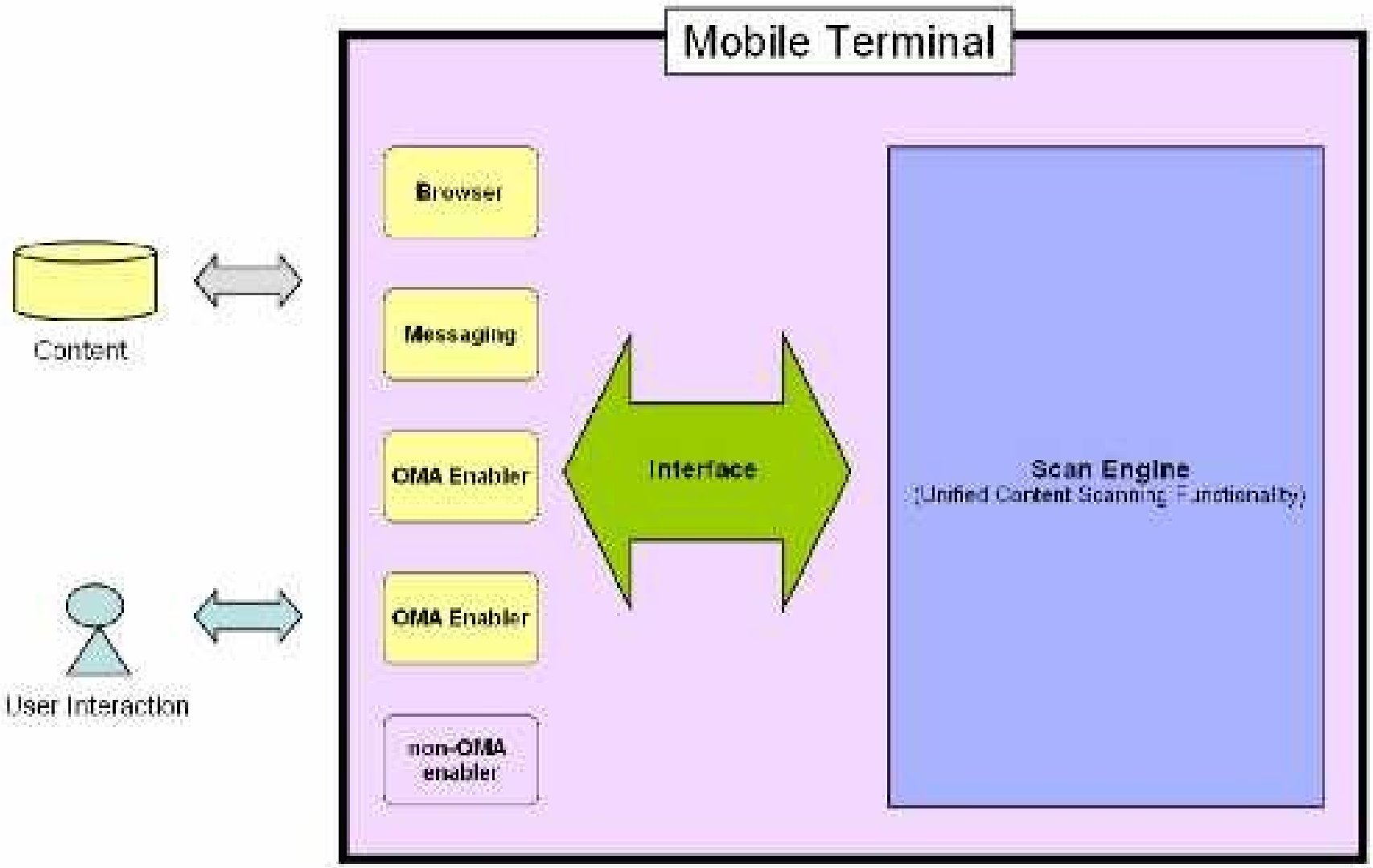
Spyware


malicious
application
Unexpected
mail sent
Address
Book
Private
Photos

Content received via end-to-end encrypted connections



CSCF Logical model



More Info:

- OMA Client Side Content Screening Framework V1.0
  - http://www.openmobilealliance.org/Technical/release_program/Client_Side_CS_FW_v1_0.aspx


McAfee®