

Infosec

Karel Kubat / karel@kubat.nl

Who am I

- Karel Kubat
- Founder, co-owner e-tunity
- Founder, owner KAOZ
- Application development & architecture
- Security consulting & architecture
- Presently with Royal Bank of Scotland, security architect for corporate banking



Audience Test

I could tell you a UDP joke,
but you might not get it.

Agenda

- Latest & greatest attacks
 - Outdated and By numbers: most often used attacks
 - Google attack & Dalai Lama's office
 - ZEUS kit - DIY
 - Certificate Authorities: Comodo, Diginotar
 - Stuxnet worm
- Challenge - Off the Grid Encryption

Outdated and By Numbers

- Outdated
 - Buffer overflow attacks
 - AnnaKournikova.jpg.exe
- By Numbers: Attacks against web sites
 - Cross Site Request Forgery
 - Cross Site Scripting
 - Content Management exploits
 - SQL injections - by far most common
 - "Soup Nazi" Albert Gonzales
 - Theft of approx. 170 mil ccards



Google & Dalai Lama's office

- Based on "live data"

- David Bonn, Linux Journal, April 1st 1996, <http://www.linuxjournal.com/article/1193>
- live data, n. 1. Data that is written to be interpreted and takes over program flow when triggered by some un-obvious operation, such as viewing it. One use of such hacks is to break security. For example, some smart terminals have commands that allow one to download strings to program keys; this can be used to write live data that, when listed to the terminal, infects it with a security-breaking virus that is triggered the next time a hapless user strikes that key. For another, there are some well-known bugs in vi that allow certain texts to send arbitrary commands back to the machine when they are simply viewed. 2. In C code, data that includes pointers to function hooks (executable code). 3. An object, such as a trampoline, that is constructed on the fly by a program and intended to be executed as code. --From the Jargon file, version 3.2.0

Live data in the 1990's

"The link seemed innocent enough. All it said was:
Crusty the Clown fans should click here!!!

Perhaps I was a bit naive—or just stupid. But I clicked on the damned link. The actual transfer didn't take very long, and pageview (Sun's PostScript viewer) quickly brought up a picture of Crusty the Clown (from The Simpsons). The first inkling of any trouble was that my disk drive light came on. And stayed on. That almost never happens, especially without me doing something. I pulled up another xterm. Something was really funny; my tcsh configuration was all wrong...

My drive light went off. My home directory was gone.

All that had happened was that someone had embedded a command line in the PostScript image. The PostScript viewer I was using happily executed the command and 200 megabytes of my files were simply erased."

Live data, current day

- Vector: Portable Document Format (pdf)
- Vulnerability: Adobe Acrobat (pdf viewer)
- Procedure: Installs executables on targeted systems
 - Personal information theft
 - E-mail scanning
 - Indexing of files
 - Theft of intellectual property (code repo)
 - Mother ship & update feature

ZEUS kit - DIY Trojans

- Framework for Windows trojan construction
- Known since 2009
 - 2010: Sold underground for \$3.000 - \$4.000
- Source code leaked in 2011
 - <http://thehackernews.com/2011/05/finally-source-code-of-zeus-crimeware.html>
- Typical usage:
 - pdf/Acrobat as vector and vulnerability
 - Man in the Browser attacks
- German state R2D2 trojan, based on ZEUS?

Man in the Browser (MITB)

- Typical: IE (provides dll hooks)
- Traffic filtering and shaping
- Key stroke logging
- Insertion of fraudulent payments
- Examples
 - ING Direct
 - Rabobank
 - RBS

Diginotar

- Bad company security
 - Attack was "amateurish"
 - Lax rules, no vault
 - 450+ certificates were created in 6 weeks
 - ... and no one noticed
- Diginotar-signed certs seen in Iran, used by approx. 300.000 ip's
- The perfect Man in the Middle attack
 - Must get in the traffic stream
 - Combine with: ISP controlled infrastructure, pseudo-WiFi access, DNS poisoning, WiFi gateway spoofing, ARP poisoning, ...

Pseudo-WiFi

- Laptop with 2 WiFi nics
- Create your own wireless network
- Be a DHCP + DNS server + gateway

- Traffic will come your way
- Harvest plain protocol streams (HTTP, IMAP, POP3, ...)
- On the fly encrypt/decrypt SSL-streams for sites that match certs

ARP cache poisoning

- Address Resolution Protocol
- ARP request: "Who is 192.168.1.254?"
- ARP response: "That's e4:ce:8f:54:f0:67"

- Spam your MAC address as gateway IP
- Traffic will come your way

DNS cache poisoning

- Kaminski attack
- Susceptible DNS servers use incremental transaction ID's
- Estimate next ID range
- Spam DNS resolutions for sites
- Traffic will come your way

Stuxnet

- To date most sophisticated attack
- Discovered June 2010
- Multiple attack vectors
 - PDF exploits, IE6 drive-by, SMB attacks, autorun.inf, many more...
- Targets PC's that control Siemens PLC's (programmable logic controllers)
- More specifically: centrifuge controllers

Stuxnet

- "Stuxnet set back Iran's nuclear program by 5 or more years."
- Successor "DuQu" seen

Challenge: Off the Grid Encryption

- Encryption algorithms date way back
 - Caesar's Cipher - substitution
 - Fairplay's Cipher - 2-letter output, distributed
- All hand-usable ciphers have been cracked
 - Brute forcing is just too easy
- Steve Gibson, <https://www.grc.com/OffTheGrid.htm>

OTG Properties

- Encrypts and decrypts only letters a-z
- Symmetric cipher
 - Security of the key is the problem
- Start with a 26x26 Latin square
 - Each letter occurs only once per row and column
 - Letters can appear in upper case for more entropy in output that is taken as a password

Sample square for A to L

H e I k g C L d f A B j
i B k G l h j c e D A f
a L D C H B I E G F J k
K A G l j I f H b c D e
f i E b a j D l H G K c
D j C h I A K B l E F g
J H f E b l A g C K I D
E k B A D f C j I L g H
G D l J f k e I a H C B
B G a d C E H f k j l I
C f H I k d G A J B e L
l C j F e g b k D I h a

OTG Algorithm

- Start cursor at given coordinate (0,0)
- Take turns looking up the next character to encrypt
 - Horizontally - left or right
 - Vertically - up or down
- When the character is found:
 - Output the two following characters
 - Advance cursor for the next round by skipping over the output characters
- Example: encrypt "hack"

Encrypt **h**

1. Find **h**
2. Output next 2 characters **eI**
3. Cursor is at the **k**

```
H e I k g C L d f A B j
i B k G l h j c e D A f
a L D C H B I E G F J k
K A G l j I f H b c D e
f i E b a j D l H G K c
D j C h I A K B l E F g
J H f E b l A g C K I D
E k B A D f C j I L g H
G D l J f k e I a H C B
B G a d C E H f k j l I
C f H I k d G A J B e L
l C j F e g b k D I h a
```

Encrypt **ha**

1. Find **h** horizontally
2. Output next 2 characters **eI**
3. Cursor is at the **k**
4. Find **a** vertically
5. Output next 2 characters **Jd**
6. Cursor is at the **l**

H	e	I	k	g	C	L	d	f	A	B	j
i	B	k	G	l	h	j	c	e	D	A	f
a	L	D	C	H	B	I	E	G	F	J	k
K	A	G	l	j	I	f	H	b	c	D	e
f	i	E	b	a	j	D	l	H	G	K	c
D	j	C	h	I	A	K	B	l	E	F	g
J	H	f	E	b	l	A	g	C	K	I	D
E	k	B	A	D	f	C	j	I	L	g	H
G	D	l	J	f	k	e	I	a	H	C	B
B	G	a	d	C	E	H	f	k	j	l	I
C	f	H	I	k	d	G	A	J	B	e	L
l	C	j	F	e	g	b	k	D	I	h	a

Encrypt **hac**

1. Find **h** horizontally
2. Output next 2 characters **eI**
3. Cursor is at the **k**
4. Find **a** vertically
5. Output next 2 characters **Jd**
6. Cursor is at the **l**
7. Find **c** horizontally
8. Output next 2 characters **Le**
9. Cursor is at the **B**

H	e	I	k	g	C	L	d	f	A	B	j
i	B	k	G	l	h	j	c	e	D	A	f
a	L	D	C	H	B	I	E	G	F	J	k
K	A	G	l	j	I	f	H	b	c	D	e
f	i	E	b	a	j	D	l	H	G	K	c
D	j	C	h	I	A	K	B	l	E	F	g
J	H	f	E	b	l	A	g	C	K	I	D
E	k	B	A	D	f	C	j	I	L	g	H
G	D	l	J	f	k	e	I	a	H	C	B
B	G	a	d	C	E	H	f	k	j	l	I
C	f	H	I	k	d	G	A	J	B	e	L
l	C	j	F	e	g	b	k	D	I	h	a

Encrypt **hack**

1. Find **h** horizontally
2. Output next 2 characters **eI**
3. Cursor is at the **k**
4. Find **a** vertically
5. Output next 2 characters **Jd**
6. Cursor is at the **l**
7. Find **c** horizontally
8. Output next 2 characters **Le**
9. Cursor is at the **B**
10. Find **k** vertically
11. Output next 2 characters **EG**
12. Cursor is at the **c**

```
H e I k g C L d f A B j
i B k G l h j c e D A f
a L D C H B I E G F J k
K A G l j I f H b c D e
f i E b a j D l H G K c
D j C h I A K B l E F g
J H f E b l A g C K I D
E k B A D f C j I L g H
G D l J f k e I a H C B
B G a d C E H f k j l I
C f H I k d G A J B e L
l C j F e g b k D I h a
```

OTG Claims

- Can be performed by hand
 - All you need is your Latin square
- Enough entropy to last you a lifetime
 - 1400+ bits of entropy
- Some leakage
 - But more than enough entropy left
- Unbreakable with current methods

