Knocking down the HACIENDA

Julian Kirsch
Advisor: Christian Grothoff

Technische Universität München

August 15, 2014

Motivation

What is HACIENDA?

- Data reconnaissance tool developed by the CITD team in JTRIG
- Port Scans entire countries
  - Uses nmap as port scanning tool
  - Uses GEOFUSION for IP Geolocation
  - Randomly scans every IP identified for that country

Countries

- Completed full scans of 27 countries including
  - [redacted]
  - [redacted]
  - [redacted]
  - [redacted]
  - [redacted]
  - [redacted]
- Completed partial scans of 5 additional countries

Tasking & Access

- To task HACIENDA with a Country or Subnet
  - [redacted]@gchq.gov.uk)
  - CITD alias ([redacted]@gchq.gov.uk)

- Access to the Data
  - At GCHQ, request a GLOBAL SURGE account from [redacted]@gchq.gov.uk)
  - At CSEC, contact [redacted]
  - At NSA, contact [redacted]
  - At DSD, contact [redacted]

[redacted]

Ports

- Pulls back hostname, banners, application names and port status
- Gathers additional information for…

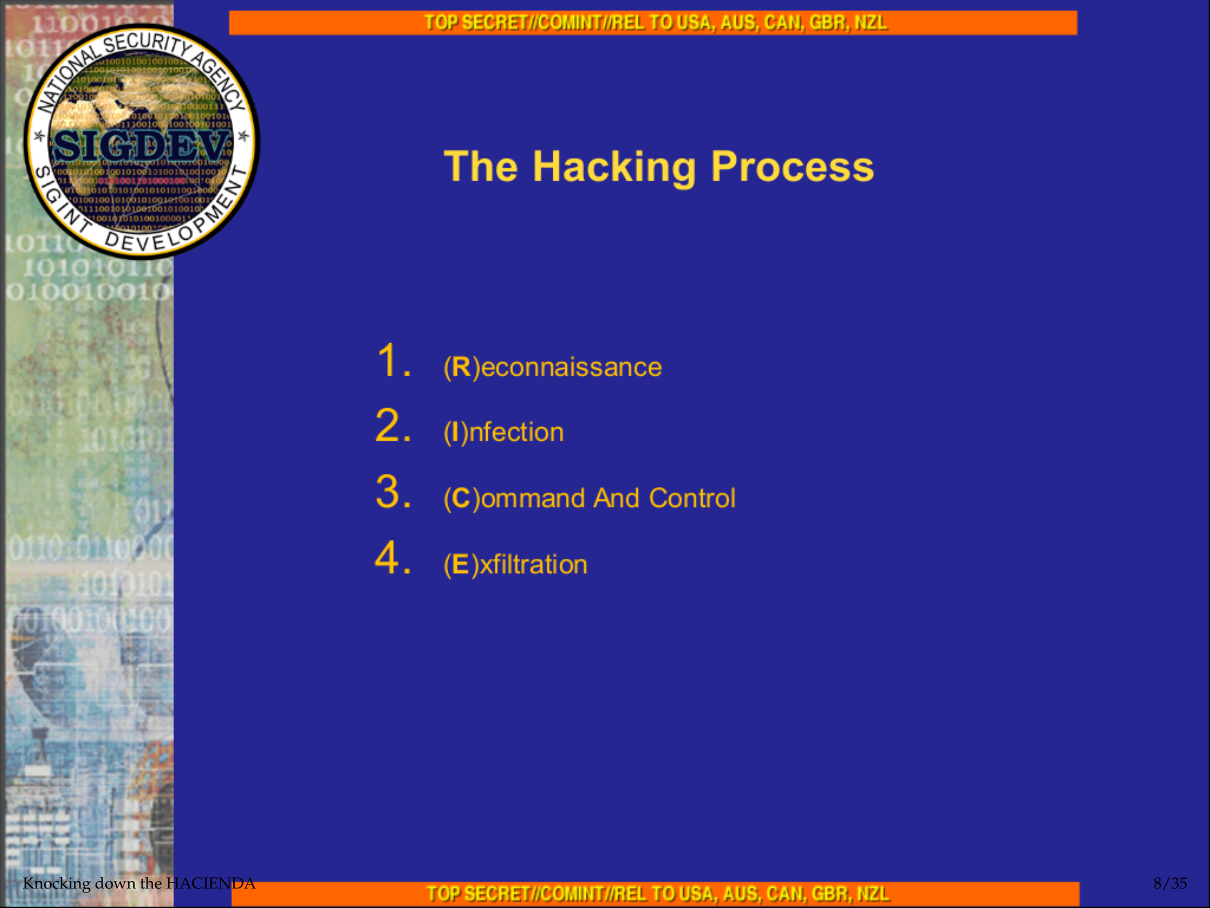
– 21 (ftp):	directory listing
– 80 (http):	content of main page
– 443 (https):	content of main page
– 111 (rpc):	results of rpcinfo

How is it used?

- CNE
  - ORB Detection
  - Vulnerability Assessments
- SD
  - Network Analysis
  - Target Discovery

Step 3

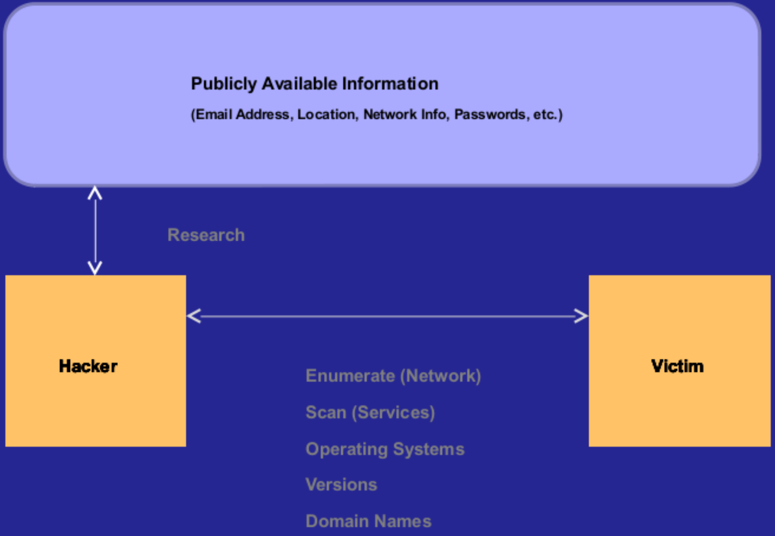
Hacking in SIGINT



The Hacking Process

1. (**R**)econnaissance
2. (**I**)nfection
3. (**C**)ommand And Control
4. (**E**)xfiltration

Reconnaissance



Publicly Available Information

(Email Address, Location, Network Info, Passwords, etc.)

Research

Hacker

Victim

Enumerate (Network)

Scan (Services)

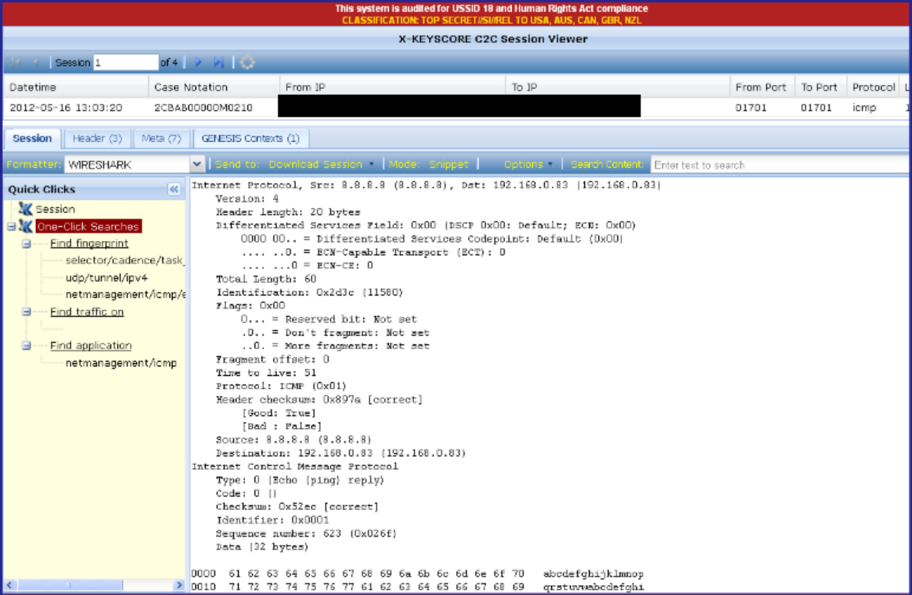
Operating Systems

Versions

Domain Names

Reconnaissance Infection Command and Control Exfiltration

Reconnaissance



This system is audited for USSID 18 and Human Rights Act compliance
CLASSIFICATION: TOP SECRET//SI//REL TO USA, AUS, CAN, GBR, NZL

X-KEYSCORE C2C Session Viewer

Datetime	Case Notation	From IP	To IP	From Port	To Port	Protocol	L
2012-05-16 13:03:20	2CBAB00000M0210	[redacted]	[redacted]	01701	01701	icmp	1

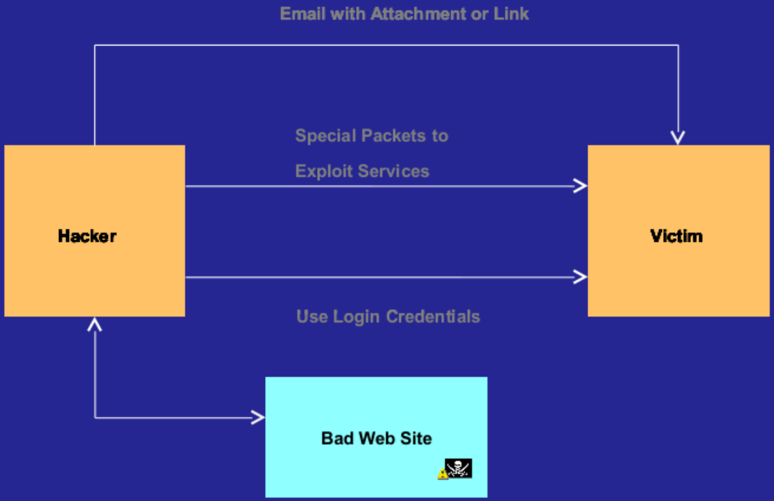
Formatter: WIRESHARK

Mode: Snippet

```
Internet Protocol, Src: 8.8.8.8 (8.8.8.8), Dst: 192.168.0.83 (192.168.0.83)
    Version: 4
    Header length: 20 bytes
    Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
        0000 00.. = Differentiated Services Codepoint: Default (0x00)
        .... ..0. = ECN-Capable Transport (ECT): 0
        .... ...0 = ECN-CE: 0
    Total Length: 60
    Identification: 0x2d3c (11580)
    Flags: 0x00
        0... = Reserved bit: Not set
        .0.. = Don't fragment: Not set
        ..0. = More fragments: Not set
    Fragment offset: 0
    Time to live: 51
    Protocol: ICMP (0x01)
    Header checksum: 0x897a [correct]
        [Good: True]
        [Bad : False]
    Source: 8.8.8.8 (8.8.8.8)
    Destination: 192.168.0.83 (192.168.0.83)
Internet Control Message Protocol
    Type: 0 (Echo (ping) reply)
    Code: 0 ()
    Checksum: 0x52ec [correct]
    Identifier: 0x0001
    Sequence number: 623 (0x026f)
    Data (32 bytes)

0000  61 62 63 64 65 66 67 68 69 6a 6b 6c 6d 6e 6f 70   abcdefghijklmnop
0010  71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69   qrstuvwabcdefghi
```

Reconnaissance Infection Command and Control Exfiltration

Infection



Email with Attachment or Link

Special Packets to Exploit Services

Hacker

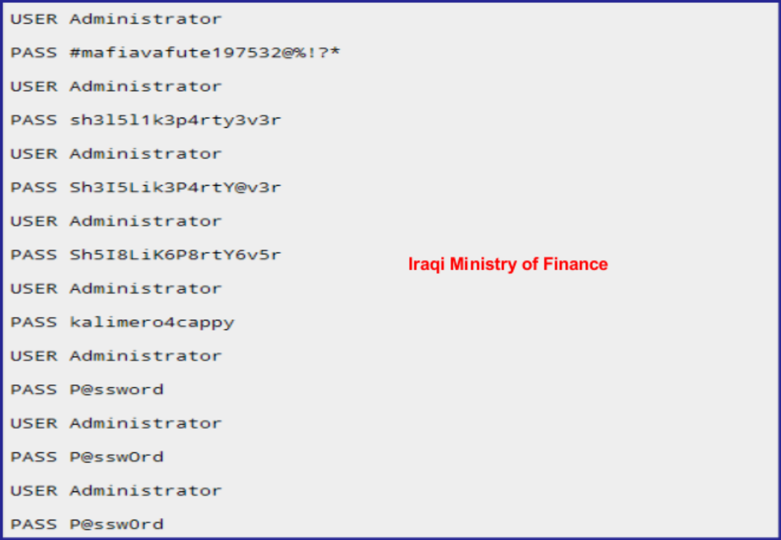
Victim

Use Login Credentials

Bad Web Site

Reconnaissance Infection Command and Control Exfiltration

Password Guessing

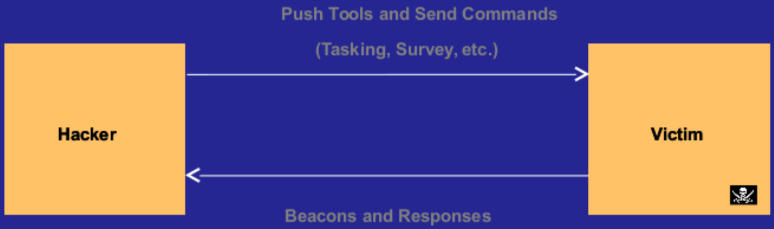


```
USER Administrator
PASS #mafiavafute197532@%!?*
USER Administrator
PASS sh3l5l1k3p4rty3v3r
USER Administrator
PASS Sh3I5Lik3P4rtY@v3r
USER Administrator
PASS Sh5I8LiK6P8rtY6v5r
USER Administrator
PASS kalimero4cappy
USER Administrator
PASS P@ssword
USER Administrator
PASS P@sswOrd
USER Administrator
PASS P@ssw0rd
```

Iraqi Ministry of Finance

Reconnaissance Infection Command and Control Exfiltration

Command and Control



Push Tools and Send Commands

(Tasking, Survey, etc.)

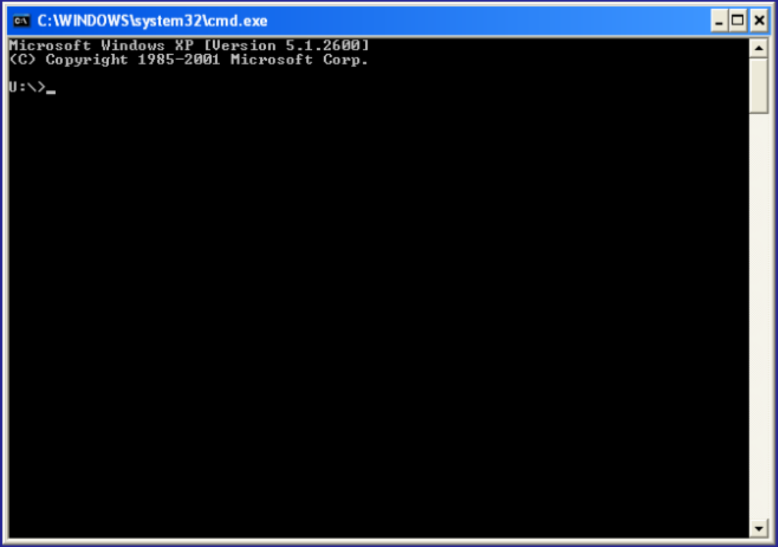
Hacker

Victim

Beacons and Responses

Reconnaissance Infection Command and Control Exfiltration

Windows cmd.exe



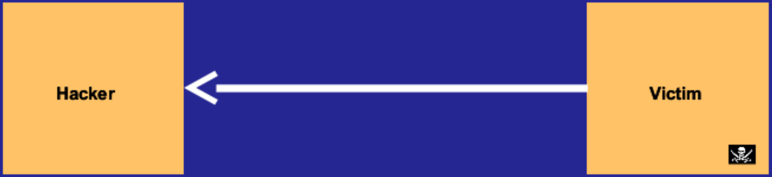
```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

U:\>_
```

Reconnaissance Infection **Command and Control** Exfiltration

Exfiltration

Exfil using known and custom protocols

(Known: HTTP, SMTP, ICMP, FTP, etc)



Reconnaissance Infection Command and Control **Exfiltration**

LANDMARK

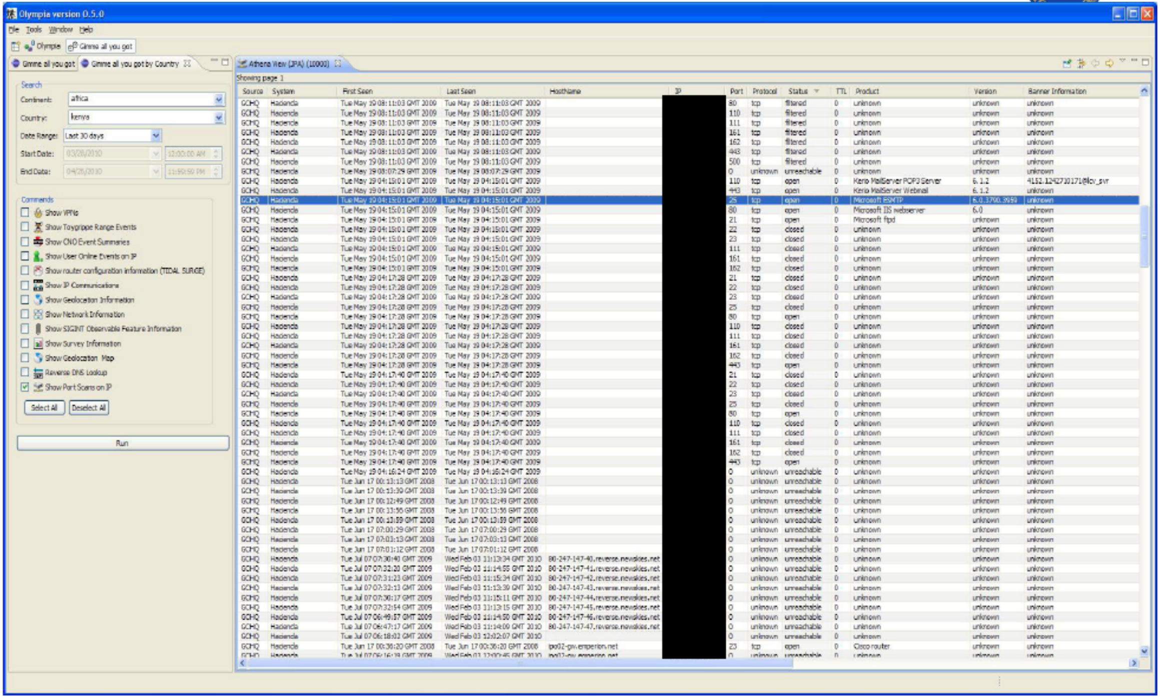
- CSEC's Operational Relay Box (ORB) covert infrastructure used to provide an additional level of non-attribution; subsequently used for exploits and exfiltration
- 2-3 times/year, 1 day focused effort to acquire as many new ORBs as possible in as many non 5-Eyes countries as possible

[redacted]

[redacted]

[redacted]

[redacted]



Olympia version 0.5.0

File Tools Window Help

Olympia | Gimme all you got

Gimme all you got | Gimme all you got by Country

Search

Continent: africa

Country: kenya

Date Range: Last 30 days

Start Date: 03/28/2010 12:00:00 AM

End Date: 04/26/2010 11:59:59 PM

Commands

- ☐ Show VPNs
- ☐ Show Toygrippe Range Events
- ☐ Show CNO Event Summaries
- ☐ Show User Online Events on IP
- ☐ Show router configuration information (TIDAL SURGE)
- ☐ Show IP Communications
- ☐ Show Geolocation Information
- ☐ Show Network Information
- ☐ Show SIGINT Observable Feature Information
- ☐ Show Survey Information
- ☐ Show Geolocation Map
- ☐ Reverse DNS Lookup
- ☑ Show Port Scans on IP

Select All | Deselect All

Run

Athena View (JPA) (10000)

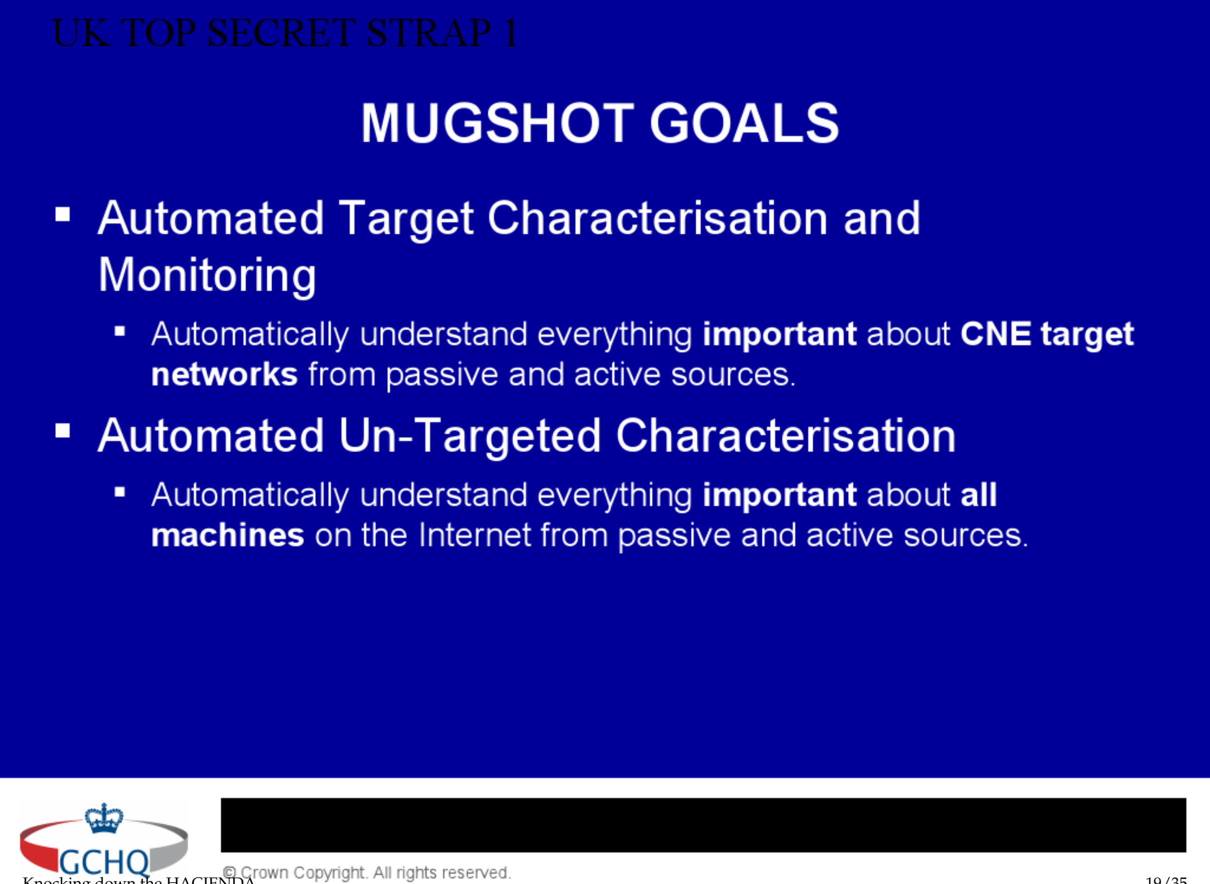
Showing page 1

Source	System	First Seen	Last Seen	HostName	IP	Port	Protocol	Status	TTL	Product	Version	Banner Information
GCHQ	Hacienda	Tue May 19 08:11:03 GMT 2009	Tue May 19 08:11:03 GMT 2009		[redacted]	80	tcp	filtered	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 08:11:03 GMT 2009	Tue May 19 08:11:03 GMT 2009		[redacted]	110	tcp	filtered	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 08:11:03 GMT 2009	Tue May 19 08:11:03 GMT 2009		[redacted]	111	tcp	filtered	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 08:11:03 GMT 2009	Tue May 19 08:11:03 GMT 2009		[redacted]	161	tcp	filtered	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 08:11:03 GMT 2009	Tue May 19 08:11:03 GMT 2009		[redacted]	162	tcp	filtered	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 08:11:03 GMT 2009	Tue May 19 08:11:03 GMT 2009		[redacted]	443	tcp	filtered	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 08:11:03 GMT 2009	Tue May 19 08:11:03 GMT 2009		[redacted]	500	tcp	filtered	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 08:07:29 GMT 2009	Tue May 19 08:07:29 GMT 2009		[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:15:01 GMT 2009	Tue May 19 04:15:01 GMT 2009		[redacted]	110	tcp	open	0	Kerio MailServer POP3 Server	6.1.2	4152.1242710171@lcv_svr
GCHQ	Hacienda	Tue May 19 04:15:01 GMT 2009	Tue May 19 04:15:01 GMT 2009		[redacted]	443	tcp	open	0	Kerio MailServer Webmail	6.1.2	unknown
GCHQ	Hacienda	Tue May 19 04:15:01 GMT 2009	Tue May 19 04:15:01 GMT 2009		[redacted]	25	tcp	open	0	Microsoft ESMTP	6.0.3790.3959	unknown
GCHQ	Hacienda	Tue May 19 04:15:01 GMT 2009	Tue May 19 04:15:01 GMT 2009		[redacted]	80	tcp	open	0	Microsoft IIS webserver	6.0	unknown
GCHQ	Hacienda	Tue May 19 04:15:01 GMT 2009	Tue May 19 04:15:01 GMT 2009		[redacted]	21	tcp	open	0	Microsoft ftpd	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:15:01 GMT 2009	Tue May 19 04:15:01 GMT 2009		[redacted]	22	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:15:01 GMT 2009	Tue May 19 04:15:01 GMT 2009		[redacted]	23	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:15:01 GMT 2009	Tue May 19 04:15:01 GMT 2009		[redacted]	111	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:15:01 GMT 2009	Tue May 19 04:15:01 GMT 2009		[redacted]	161	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:15:01 GMT 2009	Tue May 19 04:15:01 GMT 2009		[redacted]	162	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:28 GMT 2009	Tue May 19 04:17:28 GMT 2009		[redacted]	21	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:28 GMT 2009	Tue May 19 04:17:28 GMT 2009		[redacted]	22	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:28 GMT 2009	Tue May 19 04:17:28 GMT 2009		[redacted]	23	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:28 GMT 2009	Tue May 19 04:17:28 GMT 2009		[redacted]	25	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:28 GMT 2009	Tue May 19 04:17:28 GMT 2009		[redacted]	80	tcp	open	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:28 GMT 2009	Tue May 19 04:17:28 GMT 2009		[redacted]	110	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:28 GMT 2009	Tue May 19 04:17:28 GMT 2009		[redacted]	111	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:28 GMT 2009	Tue May 19 04:17:28 GMT 2009		[redacted]	161	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:28 GMT 2009	Tue May 19 04:17:28 GMT 2009		[redacted]	162	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:28 GMT 2009	Tue May 19 04:17:28 GMT 2009		[redacted]	443	tcp	open	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:40 GMT 2009	Tue May 19 04:17:40 GMT 2009		[redacted]	21	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:40 GMT 2009	Tue May 19 04:17:40 GMT 2009		[redacted]	22	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:40 GMT 2009	Tue May 19 04:17:40 GMT 2009		[redacted]	23	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:40 GMT 2009	Tue May 19 04:17:40 GMT 2009		[redacted]	25	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:40 GMT 2009	Tue May 19 04:17:40 GMT 2009		[redacted]	80	tcp	open	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:40 GMT 2009	Tue May 19 04:17:40 GMT 2009		[redacted]	110	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:40 GMT 2009	Tue May 19 04:17:40 GMT 2009		[redacted]	111	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:40 GMT 2009	Tue May 19 04:17:40 GMT 2009		[redacted]	161	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:40 GMT 2009	Tue May 19 04:17:40 GMT 2009		[redacted]	162	tcp	closed	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:17:40 GMT 2009	Tue May 19 04:17:40 GMT 2009		[redacted]	443	tcp	open	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue May 19 04:16:24 GMT 2009	Tue May 19 04:16:24 GMT 2009		[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jun 17 00:13:13 GMT 2008	Tue Jun 17 00:13:13 GMT 2008		[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jun 17 00:13:39 GMT 2008	Tue Jun 17 00:13:39 GMT 2008		[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jun 17 00:12:49 GMT 2008	Tue Jun 17 00:12:49 GMT 2008		[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jun 17 00:13:56 GMT 2008	Tue Jun 17 00:13:56 GMT 2008		[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jun 17 00:13:59 GMT 2008	Tue Jun 17 00:13:59 GMT 2008		[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jun 17 07:00:29 GMT 2008	Tue Jun 17 07:00:29 GMT 2008		[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jun 17 07:03:13 GMT 2008	Tue Jun 17 07:03:13 GMT 2008		[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jun 17 07:01:12 GMT 2008	Tue Jun 17 07:01:12 GMT 2008		[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jul 07 07:30:40 GMT 2009	Wed Feb 03 11:13:34 GMT 2010	80-247-147-40.reverse.nevskies.net	[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jul 07 07:32:20 GMT 2009	Wed Feb 03 11:14:55 GMT 2010	80-247-147-41.reverse.nevskies.net	[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jul 07 07:31:23 GMT 2009	Wed Feb 03 11:15:34 GMT 2010	80-247-147-42.reverse.nevskies.net	[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jul 07 07:32:13 GMT 2009	Wed Feb 03 11:13:39 GMT 2010	80-247-147-43.reverse.nevskies.net	[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jul 07 07:30:17 GMT 2009	Wed Feb 03 11:15:11 GMT 2010	80-247-147-44.reverse.nevskies.net	[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jul 07 07:32:54 GMT 2009	Wed Feb 03 11:13:15 GMT 2010	80-247-147-45.reverse.nevskies.net	[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jul 07 06:49:57 GMT 2009	Wed Feb 03 11:14:50 GMT 2010	80-247-147-46.reverse.nevskies.net	[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jul 07 06:47:17 GMT 2009	Wed Feb 03 11:14:09 GMT 2010	80-247-147-47.reverse.nevskies.net	[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jul 07 06:18:02 GMT 2009	Wed Feb 03 12:02:07 GMT 2010		[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown
GCHQ	Hacienda	Tue Jun 17 00:36:20 GMT 2008	Tue Jun 17 00:36:20 GMT 2008	ipo02-gw.emperion.net	[redacted]	23	tcp	open	0	Cisco router	unknown	unknown
GCHQ	Hacienda	Tue Jul 07 06:16:19 GMT 2009	Wed Feb 03 12:00:45 GMT 2010	ipo02-gw.emperion.net	[redacted]	0	unknown	unreachable	0	unknown	unknown	unknown

BUT, network analysis still manual!

[redacted]

- [redacted] GSM provider
- NSA TAO requested assistance gaining access to the network
- Network analysis using OLYMPIA:
  - DNS query to determine IP address
  - IP address to network range
  - Network range to port scan
  - Are there any vulnerable devices in that range?
- Duration: < 5 minutes



UK TOP SECRET STRAP 1

MUGSHOT GOALS

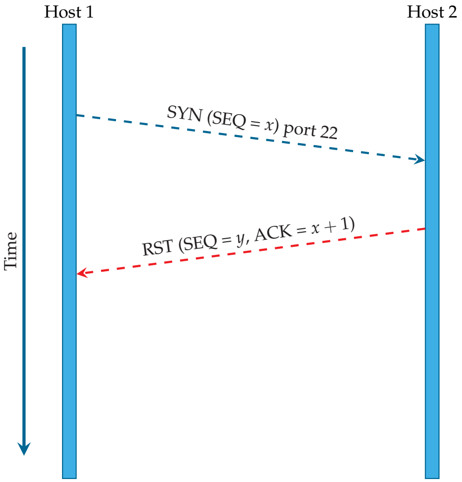
- Automated Target Characterisation and Monitoring
  - Automatically understand everything **important** about **CNE target networks** from passive and active sources.
- Automated Un-Targeted Characterisation
  - Automatically understand everything **important** about **all machines** on the Internet from passive and active sources.

[redacted]

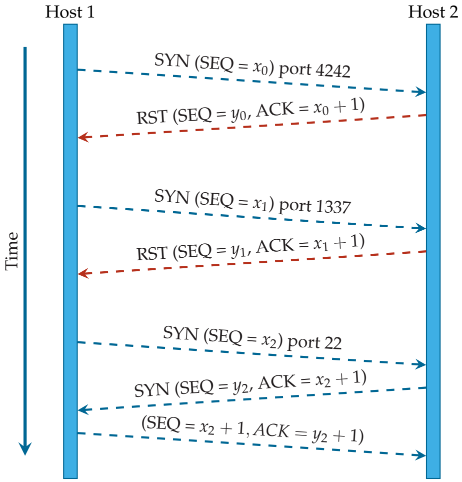
So, is it all lost?

An Introduction to Port Knocking

No knock, no fun

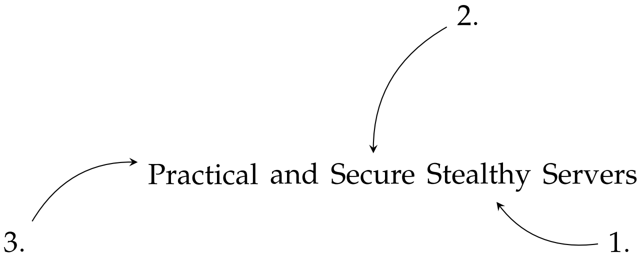


Port knocking example


Host 1
Host 2
Time
SYN (SEQ = x_0) port 4242
RST (SEQ = y_0, ACK = $x_0 + 1$)
SYN (SEQ = x_1) port 1337
RST (SEQ = y_1, ACK = $x_1 + 1$)
SYN (SEQ = x_2) port 22
SYN (SEQ = y_2, ACK = $x_2 + 1$)
(SEQ = $x_2 + 1$, ACK = $y_2 + 1$)

Design

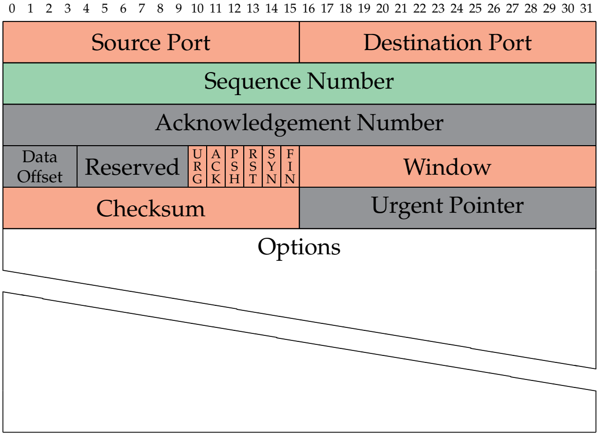
Overview



Practical and Secure Stealthy Servers

Design

Stealthiness



0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

Source Port | Destination Port

Sequence Number

Acknowledgement Number

Data Offset | Reserved | URG | ACK | PSH | RST | SYN | FIN | Window

Checksum | Urgent Pointer

Options

Design (SilentKnock)

Security

- Destination IP address IP_d
- Destination port P_d
- TCP timestamp T

- Pre-Shared Key S
- Hash function h

Authentication Security Token AV

$$\text{AV} := h((IP_d, P_d, T), S)$$

- ISN := AV

Design

Security

- Destination IP address IP_d
- Destination port P_d
- TCP timestamp T

- Pre-Shared Key S
- Hash functions h, h'
- Payload p

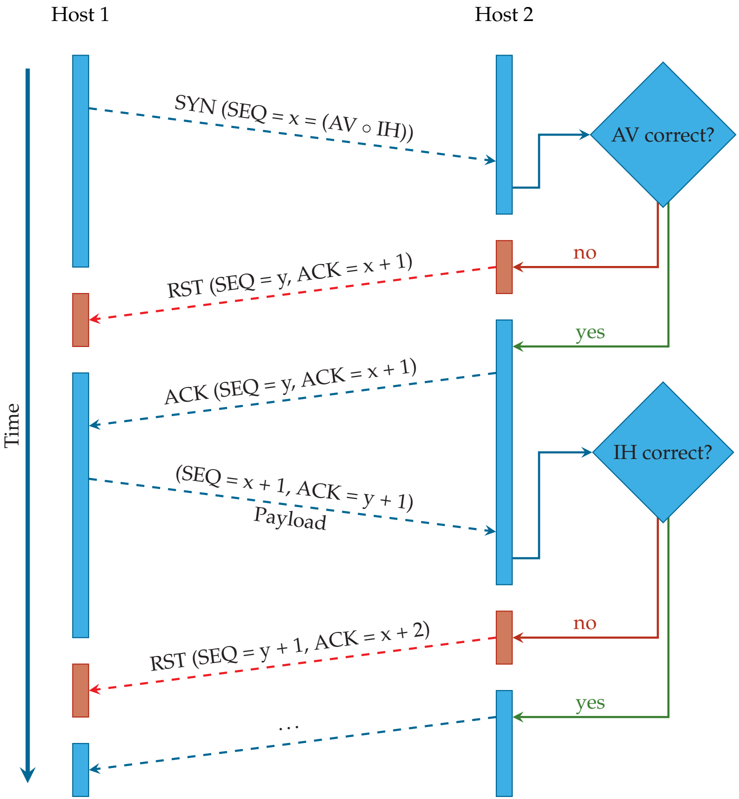
TCP Payload Integrity Protector IH

$\text{IH} := h'(S \circ p)$

Authentication Security Token AV

$\text{AV} := h((IP_d, P_d, T, \text{IH}), S)$

- $\text{ISN} := \text{AV} \circ \text{IH}$


Host 1
Host 2
Time
SYN (SEQ = x = (AV ∘ IH))
AV correct?
no
RST (SEQ = y, ACK = x + 1)
yes
ACK (SEQ = y, ACK = x + 1)
(SEQ = x + 1, ACK = y + 1)
Payload
IH correct?
no
RST (SEQ = y + 1, ACK = x + 2)
yes
. . .

Design

Ease of Use

- Source IP and Port *not* included in ISN generation
  ⇒ Compatibility with NATs
- Knocking is implemented *in the kernel*
  ⇒ No fiddling with config-files, firewall rules or daemons
  ⇒ Trivial to use from an application developer's perspective

Design

Ease of Use – TCP Stealth Server

```c
char secret[64] = "This is my magic ID.";
int payload_len = 4;
int sock;

sock = socket(AF_INET, SOCK_STREAM, IPPROTO_TCP);
if (sock < 0) {
        printf("socket() failed, %s\n", strerror(errno));
        return 1;
}
if (setsockopt(sock, IPPROTO_TCP, TCP_STEALTH, secret, sizeof(secret))) {
        printf("setsockopt() failed, %s\n", strerror(errno));
        return 1;
}
if (setsockopt(sock, IPPROTO_TCP, TCP_STEALTH_INTEGRITY_LEN,
                &payload_len, sizeof(payload_len))) {
        printf("setsockopt() failed, %s\n", strerror(errno));
        return 1;
}
/* Continue with bind(), listen(), accept(), recv(), ... */
```

Design

Ease of Use – TCP Stealth Client

```c
char secret[64] = "This is my magic ID.";
char payload[4] = "1234";
int sock;

sock = socket(AF_INET, SOCK_STREAM, IPPROTO_TCP);
if (sock < 0) {
        printf("socket() failed, %s\n", strerror(errno));
        return 1;
}
if (setsockopt(sock, IPPROTO_TCP, TCP_STEALTH, secret, sizeof(secret))) {
        printf("setsockopt() failed, %s\n", strerror(errno));
        return 1;
}
if (setsockopt(sock, IPPROTO_TCP, TCP_STEALTH_INTEGRITY,
                payload, sizeof(payload))) {
        printf("setsockopt() failed, %s\n", strerror(errno));
        return 1;
}
/* Continue with connect(), send(), ... */
```

Design

Ease of Use – libknockify

- Shared library for use at compile- or run-time
- Enables TCP Stealth functionality for legacy code

```
$ LD_PRELOAD=./libknockify.so ncat knock-server application-port
```

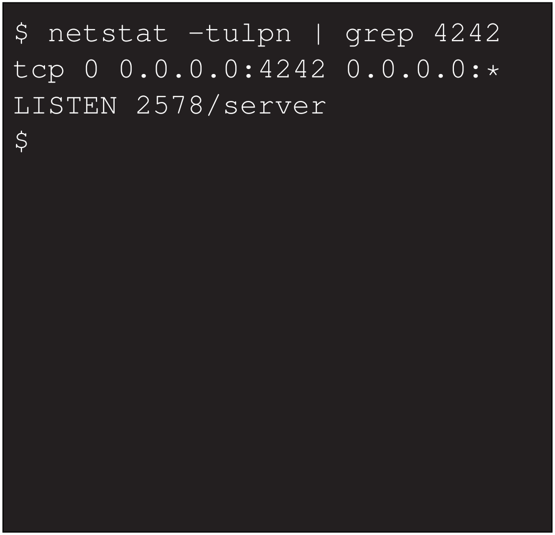
- Configuration options (such as the TCP Stealth secret) are given as environment variables or via a special file

Demo

```
$ ./server
```

Demo

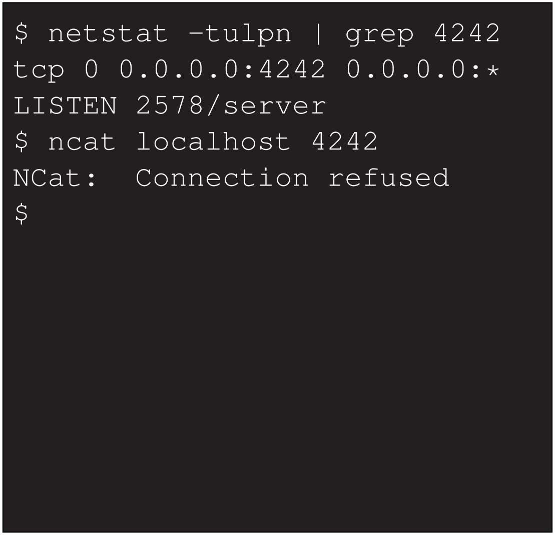
```
$ ./server
```



```
$ netstat -tulpn | grep 4242
tcp 0 0.0.0.0:4242 0.0.0.0:*
LISTEN 2578/server
$
```

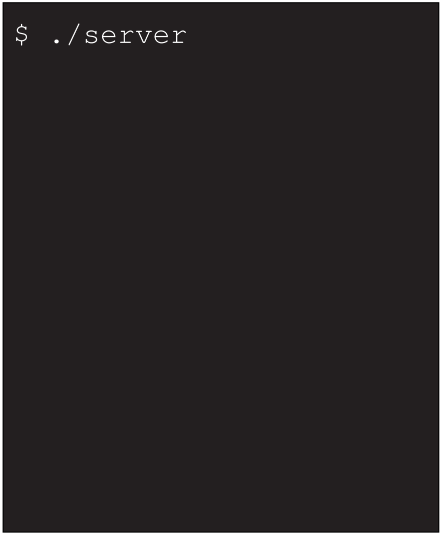
Demo

```
$ ./server
```

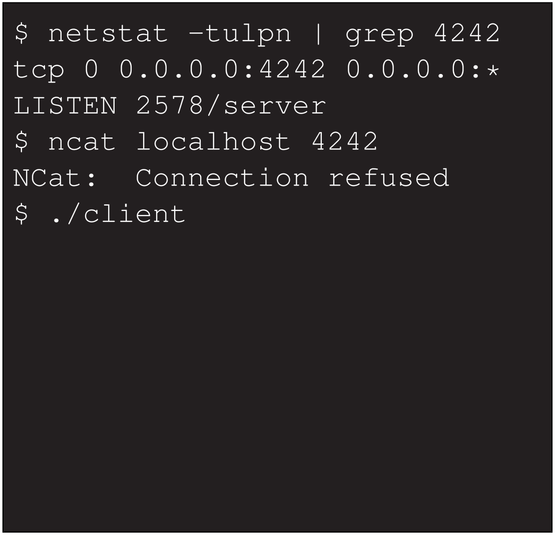


```
$ netstat -tulpn | grep 4242
tcp 0 0.0.0.0:4242 0.0.0.0:*
LISTEN 2578/server
$ ncat localhost 4242
NCat:  Connection refused
$
```

Demo



```
$ ./server
```



```
$ netstat -tulpn | grep 4242
tcp 0 0.0.0.0:4242 0.0.0.0:*
LISTEN 2578/server
$ ncat localhost 4242
NCat:  Connection refused
$ ./client
```

Demo

```
$ ./server
```

```
$ netstat -tulpn | grep 4242
tcp 0 0.0.0.0:4242 0.0.0.0:*
LISTEN 2578/server
$ ncat localhost 4242
NCat:  Connection refused
$ ./client
hello world
```

Demo

```
$ ./server
Peer closed connection.
$
```

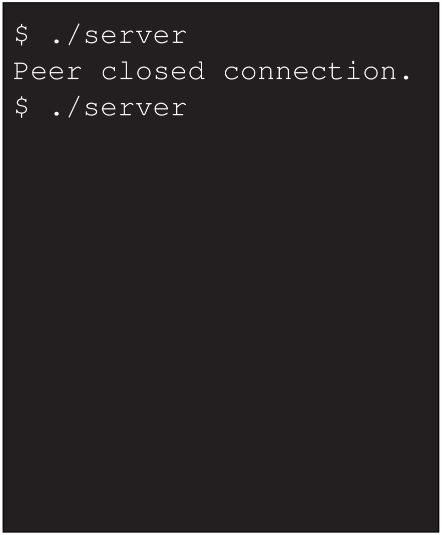
```
$ netstat -tulpn | grep 4242
tcp 0 0.0.0.0:4242 0.0.0.0:*
LISTEN 2578/server
$ ncat localhost 4242
NCat:  Connection refused
$ ./client
hello world
Peer closed connection.
$
```

Demo

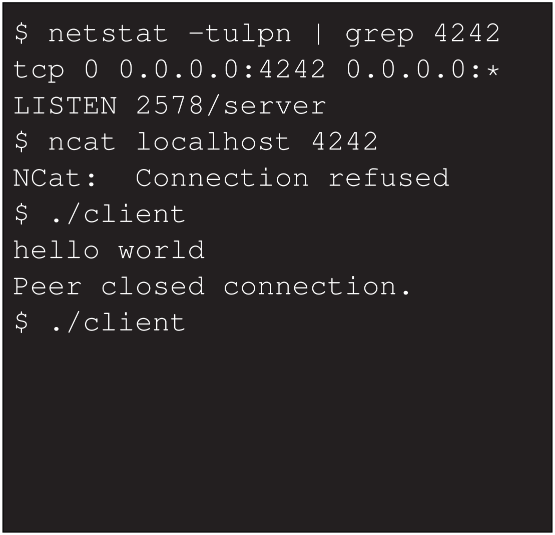
```
$ ./server
Peer closed connection.
$ ./server
```

```
$ netstat -tulpn | grep 4242
tcp 0 0.0.0.0:4242 0.0.0.0:*
LISTEN 2578/server
$ ncat localhost 4242
NCat:  Connection refused
$ ./client
hello world
Peer closed connection.
$
```

Demo

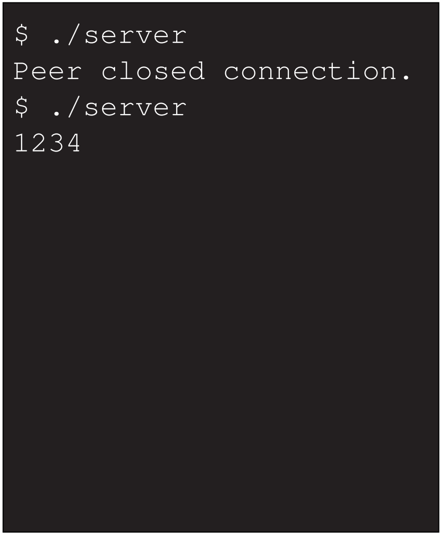


```
$ ./server
Peer closed connection.
$ ./server
```

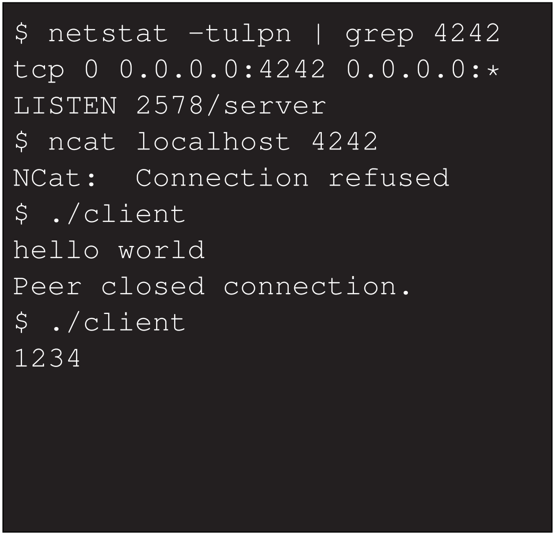


```
$ netstat -tulpn | grep 4242
tcp 0 0.0.0.0:4242 0.0.0.0:*
LISTEN 2578/server
$ ncat localhost 4242
NCat:  Connection refused
$ ./client
hello world
Peer closed connection.
$ ./client
```

Demo



```
$ ./server
Peer closed connection.
$ ./server
1234
```



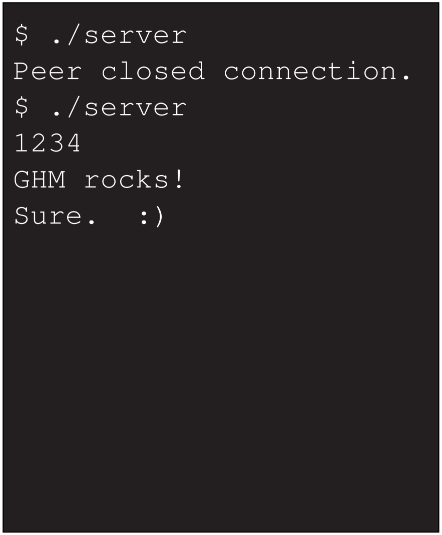
```
$ netstat -tulpn | grep 4242
tcp 0 0.0.0.0:4242 0.0.0.0:*
LISTEN 2578/server
$ ncat localhost 4242
NCat:  Connection refused
$ ./client
hello world
Peer closed connection.
$ ./client
1234
```

Demo

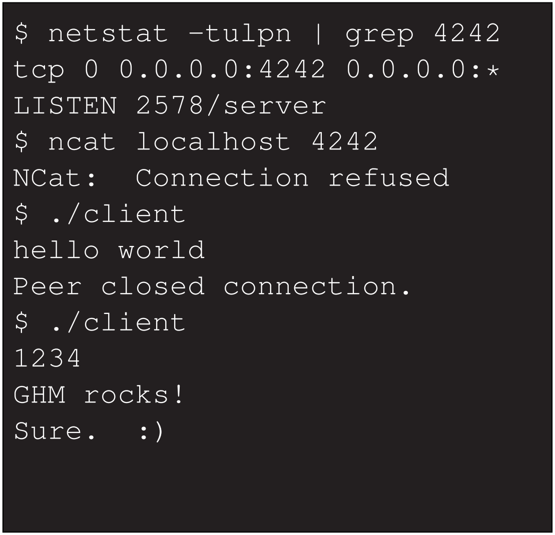
```
$ ./server
Peer closed connection.
$ ./server
1234
GHM rocks!
```

```
$ netstat -tulpn | grep 4242
tcp 0 0.0.0.0:4242 0.0.0.0:*
LISTEN 2578/server
$ ncat localhost 4242
NCat:  Connection refused
$ ./client
hello world
Peer closed connection.
$ ./client
1234
GHM rocks!
```

Demo

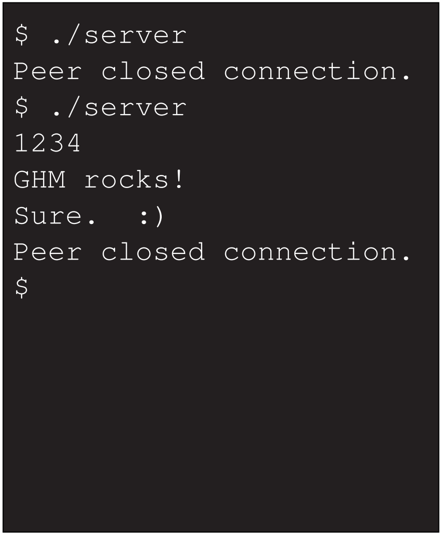


```
$ ./server
Peer closed connection.
$ ./server
1234
GHM rocks!
Sure.  :)
```

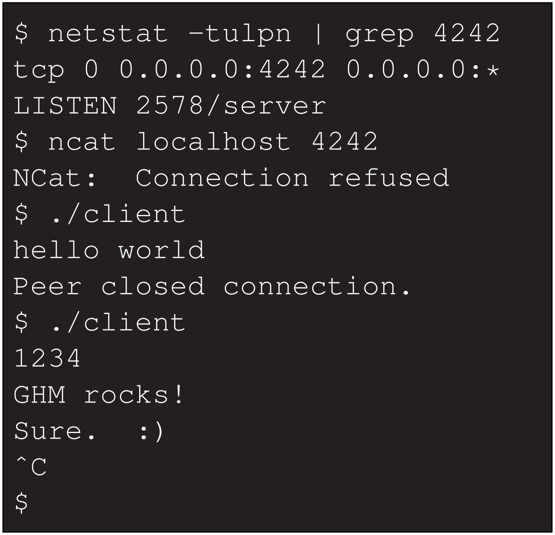


```
$ netstat -tulpn | grep 4242
tcp 0 0.0.0.0:4242 0.0.0.0:*
LISTEN 2578/server
$ ncat localhost 4242
NCat:  Connection refused
$ ./client
hello world
Peer closed connection.
$ ./client
1234
GHM rocks!
Sure.  :)
```

Demo



```
$ ./server
Peer closed connection.
$ ./server
1234
GHM rocks!
Sure.  :)
Peer closed connection.
$
```



```
$ netstat -tulpn | grep 4242
tcp 0 0.0.0.0:4242 0.0.0.0:*
LISTEN 2578/server
$ ncat localhost 4242
NCat:  Connection refused
$ ./client
hello world
Peer closed connection.
$ ./client
1234
GHM rocks!
Sure.  :)
^C
$
```

Limitations

- Distribution of the Pre-Shared Key
- ISN has only 32 bits

Acknowledgements

Christian Grothoff
Jacob Appelbaum
Monika Ermert
Laura Poitras
Henrik Moltke
Maurice Leclaire
Andreas Enge
Bart Polot
Luca Saiu
The Source

This work was funded by the Deutsche Forschungsgemeinschaft (DFG) under ENP GR 3688/1-1.

More Information

Find more information at:

https://gnunet.org/knock

https://heise.de

http://datatracker.ietf.org/doc/draft-kirsch-ietf-tcp-stealth/

End

Questions?

Thank you for your attention!

Algorithm

```
Require: $P_d$, $\mathrm{IP_d}$ in network byte order $\wedge$
  $\mathrm{len} \neq 0 \wedge \mathrm{payload}[0:\mathrm{len}] \neq 0 \wedge \mathrm{secret}[0:63] \neq 0$
Ensure: ISN in network byte order
  if $\nexists T$ then
    $T \Leftarrow 0$
  end if
  $I[0:15] \Leftarrow \mathrm{MD5}(\mathrm{secret}[0:64] \circ \mathrm{payload}[0:\mathrm{len}])$
  $\mathrm{IH}[0:1] \Leftarrow I[0:1] \oplus I[2:3] \oplus I[4:5] \oplus I[6:7] \oplus I[8:9] \oplus I[10:11] \oplus I[12:13] \oplus I[14:15]$
  if network layer is IPv4 then
    $\mathrm{IV}[0:3] \Leftarrow \mathrm{IP_d}[0:3]$
    $\mathrm{IV}[4:15] \Leftarrow 0$
  else
    if network layer is IPv6 then
      $\mathrm{IV}[0:15] \Leftarrow \mathrm{IP_d}[0:15]$
    end if
  end if
  $\mathrm{IV}[4:5] \Leftarrow \mathrm{IV}[4:5] \oplus \mathrm{IH}[0:1]$
  $\mathrm{IV}[8:11] \Leftarrow \mathrm{IV}[8:11] \oplus T$
  $\mathrm{IV}[12:13] \Leftarrow \mathrm{IV}[12:13] \oplus P_d$
  $\mathrm{AV}[0:15] \Leftarrow \mathrm{MD5Transform}(\mathrm{IV}[0:15], \mathrm{secret}[0:63])$
  $\mathrm{AV}[0:3] \Leftarrow \mathrm{AV}[0:3] \oplus \mathrm{AV}[4:7] \oplus \mathrm{AV}[8:11] \oplus \mathrm{AV}[12:15]$
  return $\mathrm{AV}[0:1] \circ \mathrm{IH}[0:1]$
```

Changes to ISN and TSVal by middle boxes

	TCP Port		
Behavior	34343	80	443
Unchanged	126 (93%)	116 (82%)	128 (90%)
Mod. outbound	5 (4%)	5 (4%)	6 (4%)
Mod. inbound	0 (0%)	1 (1%)	1 (1%)
Mod. both	4 (3%)	13 (9%)	7 (5%)
Proxy (probably mod. both)	0 (0%)	7 (5%)	0 (0%)
Total	135 (100%)	142 (100%)	142 (100%)

Numbers by Honda et al. “Is it Still Possible to Extend TCP?”