UNIVERSITY OF CAMBRIDGE

Bitter Harvest: Systematically Fingerprinting Low- and Medium-interaction Honeypots at Internet Scale

Alexander Vetterl and Richard Clayton

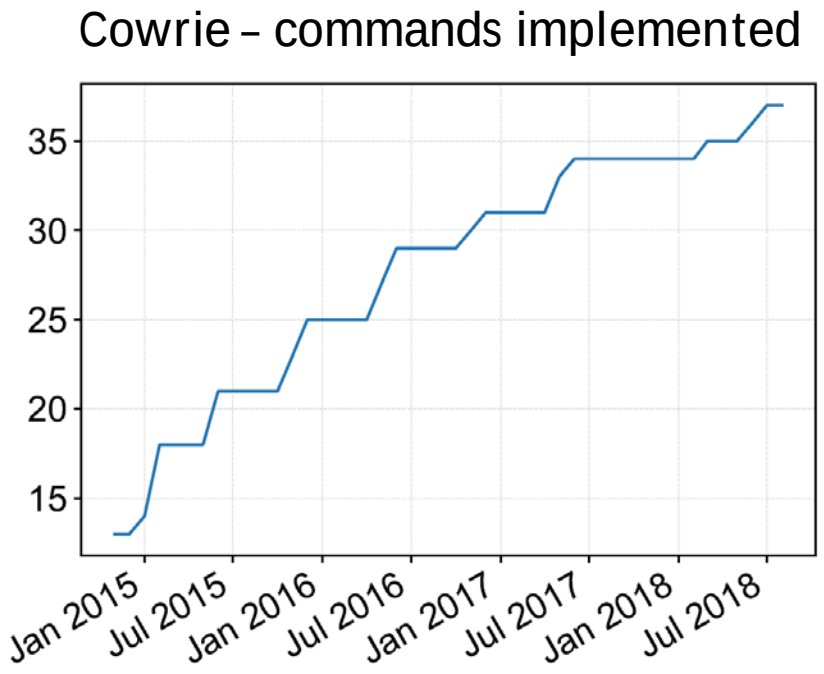
University of Cambridge

12th USENIX Workshop on Offensive Technologies – August 13-14, 2018

Introduction

Honeypots:
A resource whose value is being attacked or compromised

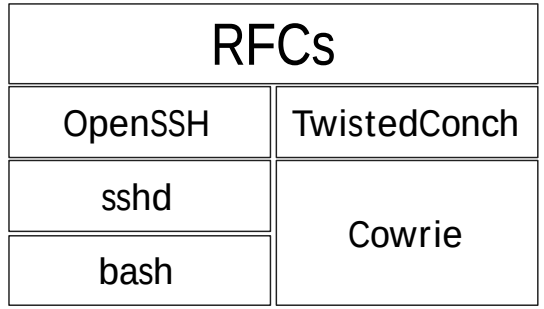
- Honeypots have been focused for years on the monitoring of human activity
- Adversaries attempt to distinguish honeypots by executing commands
- Honeypots continuously fix commands to be “more like bash”



How we currently build (SSH) honeypots

1. Find a library that implements the desired protocol (e.g. TwistedConch for SSH)
2. Write the Python program to be “just like bash”
3. Fix identity strings, error messages etc. to be “just like OpenSSH”

```
def _unsupportedVersionReceived(self, remoteVersion):
    """
    Change message to be like OpenSSH
    """
    self.transport.write(b'Protocol major versions differ.\n')
```

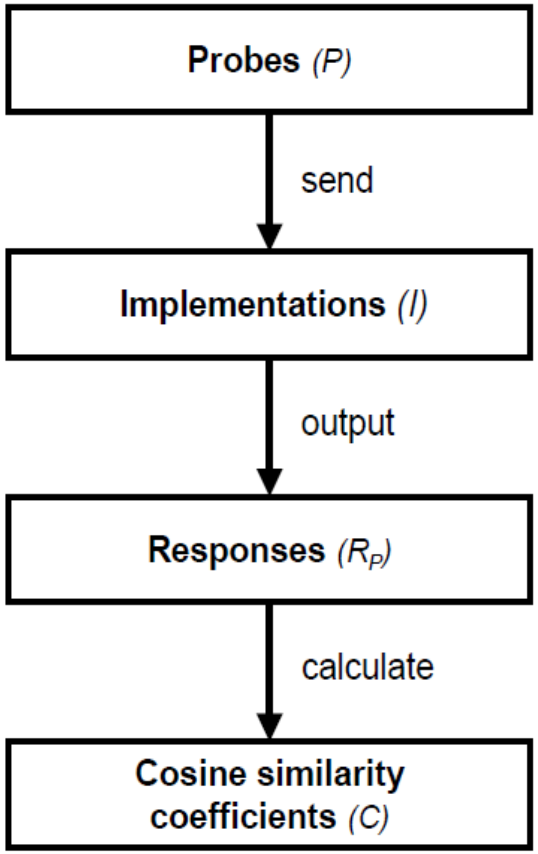


<u>Problem:</u>
There are lot of subtle differences between TwistedConch and OpenSSH!

Honeypots in this study

	Updated	Language	Library
SSH			
Kippo	May 15	Python	TwistedConch
Cowrie	May 18	Python	TwistedConch
Telnet			
TPwd	Feb 16	C	custom
MTPot	Mar 17	Python	telnetsrv
TIoT	May 17	Python	custom
Cowrie	May 18	Python	TwistedConch
HTTP/Web			
Dionaea	Sep 16	Python	custom
Glastopf	Oct 16	Python	BaseHTTPServer
Conpot	Mar 18	Python	BaseHTTPServer

Methodology – Overview



We send probes to 40 different implementations

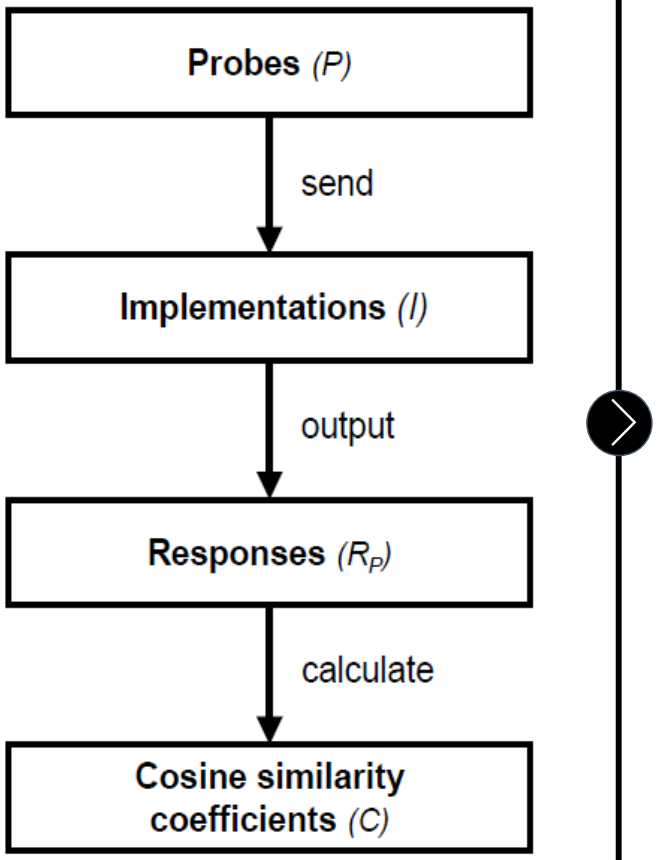
- 9 Honeypots
- OpenSSH, TwistedConch
- Busybox, Ubuntu/FreeBSD telnetd
- Apache, nginx

We find probes that result in distinctive responses

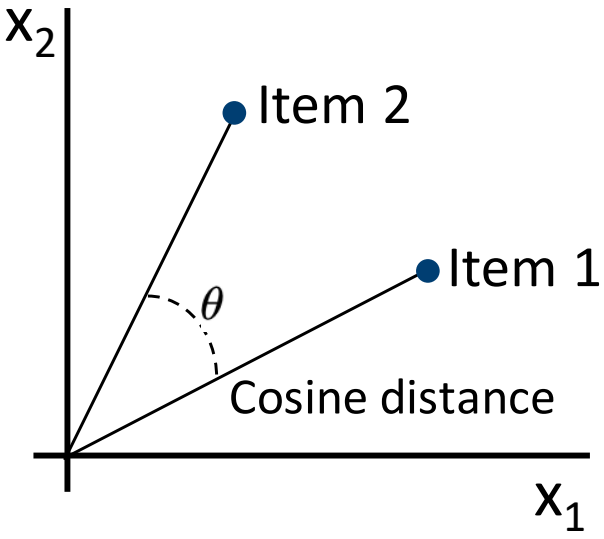
We find ‘the’ probe that results in the most distinctive response across all implementations and perform Internet wide scans

→ Triggered 158 million responses

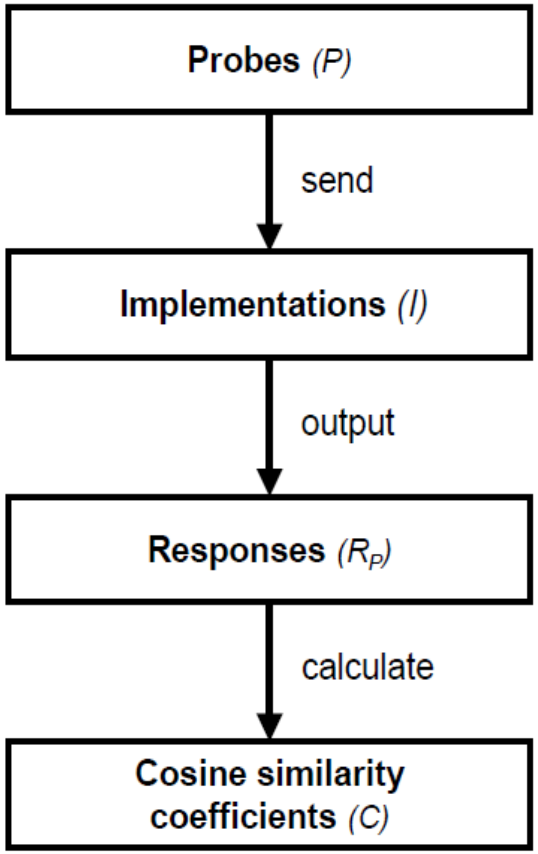
Methodology – Cosine similarity



- We represent our responses as a vector of features appropriate to the network protocol
- The higher the cosine similarity coefficient, the more similar the two items under comparison



Probe generation – Telnet and HTTP



25 440 Telnet negotiation sequences (RFC854)

4 option codes (WILL, WON'T, DO, DON'T)

IAC WILL BINARY IAC WILL LOGOUT

IAC escape character — 40 Telnet options

47 600 HTTP requests (RFC2616 and RFC2518)

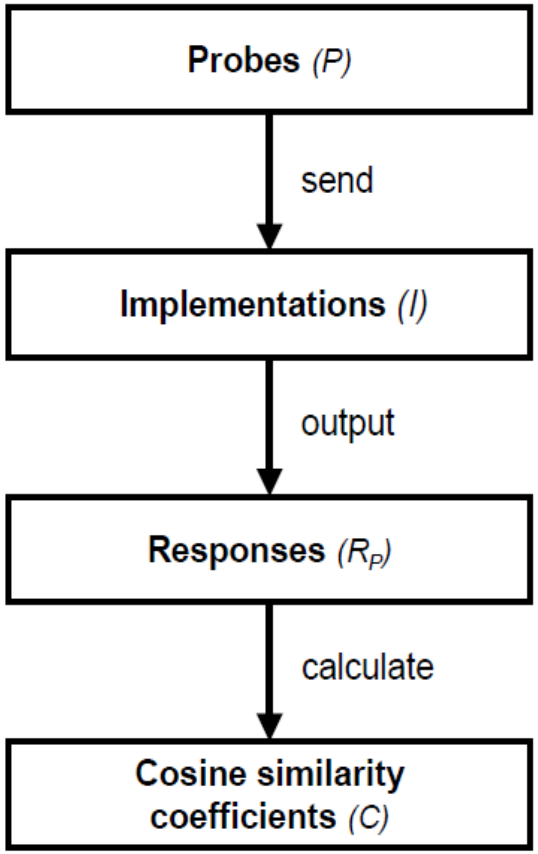
43 different request methods

GET /. HTTP/0.0.\r\n\r\n

123 non-printable, non-alphanumeric characters

9 different HTTP versions (HTTP/0.0 to HTTP/2.2)

Probe generation – SSH



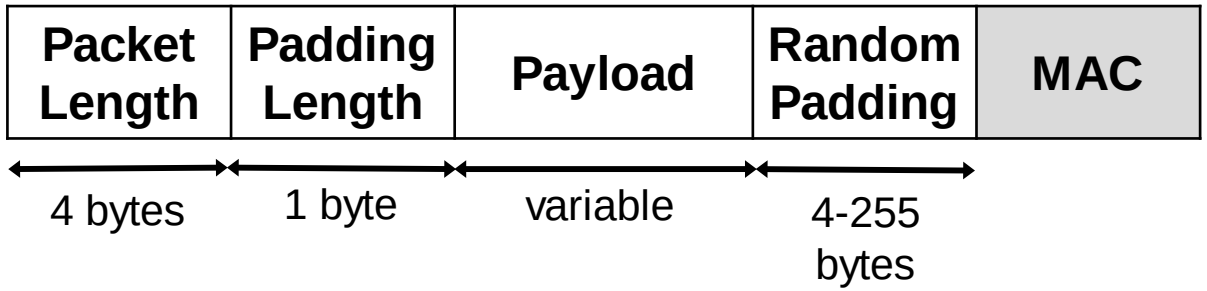
192 SSH version strings (RFC4253)

- [SSH, ssh]-[0.0 – 3.2]-[OpenSSH, ""] SP [FreeBSD, ""][\r\n, ""]

58 752 KEX_INIT packets (RFC4250)

- 16 key-exchange algorithms, 2 host key algorithms
- 15 encryption algorithms, 5 MAC algorithms,
- 3 compression algorithms

Three variants of (malformed) packets



Results – Similarity across implementations

SSH

n=157 925 376

	OpenSSH					Twisted
	6.6	6.7	6.8	7.2	7.5	15.2.1
Kippo	0.75	0.76	0.76	0.76	0.80	0.56
Cowrie	0.78	0.80	0.78	0.80	0.78	0.50

Telnet

n=356 160

	Busybox 1.6.1-2.6.2	FreeBSD 11.1 telnetd	Ubuntu 16.04 telnetd
MTPot	0.89	0.89	0.86
Cowrie	0.83	0.97	0.94
TPwd	0.89	0.87	0.85
TIoT	0.85	0.94	0.96

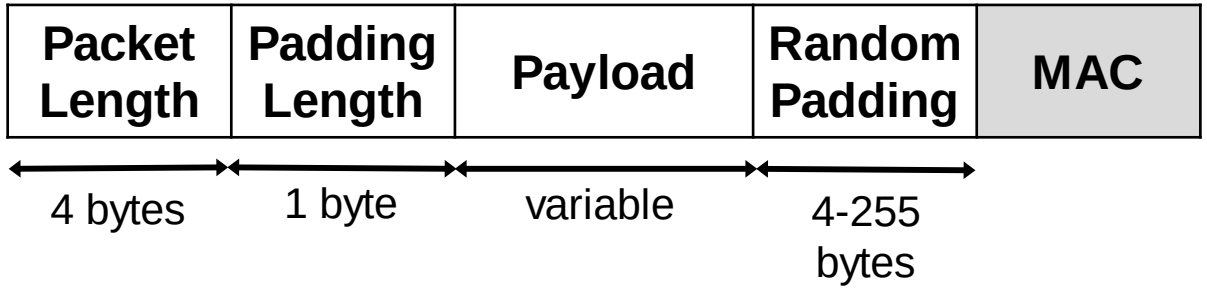
HTTP

n=571 212

	Apache			nginx		
	2.0.50	2.2.34	2.4.27	1.12.1	1.4.7	1.0.15
Glastopf	0.02	0.01	<0.01	<0.01	<0.01	<0.01
Conpot	0.10	0.09	0.09	0.04	0.02	0.02
Dionaea	0.19	0.20	0.20	0.17	0.10	0.11

Results – Reasons for distinctive responses

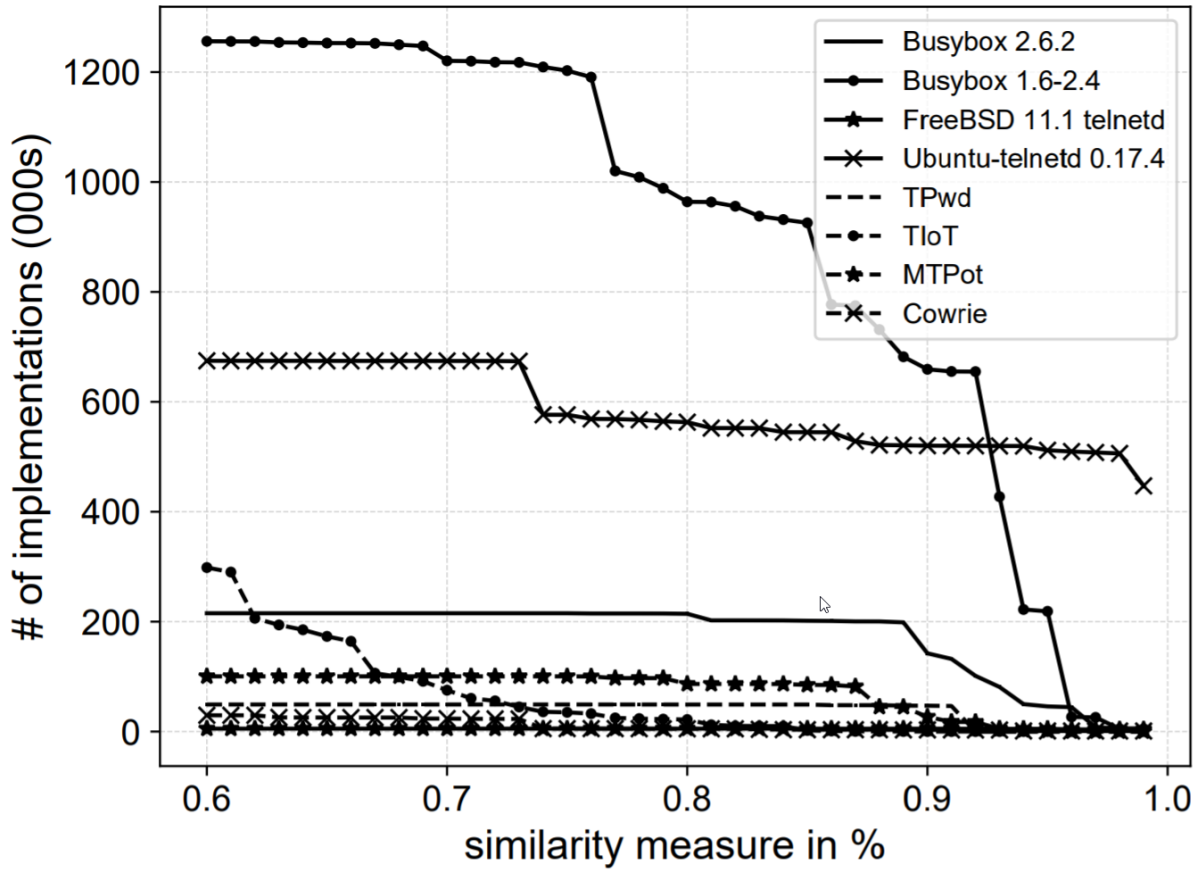
- (Random) padding of SSH packets



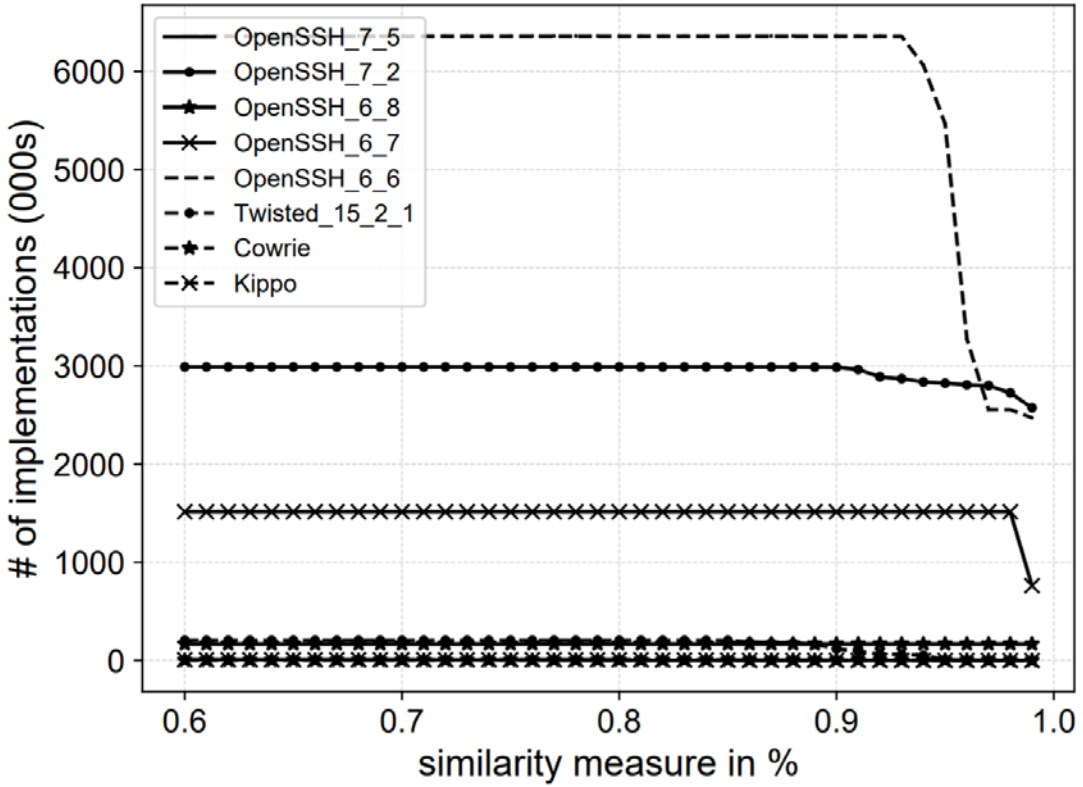
- Servers close the connection as a result of bad packets
- Not supported or ignored HTTP methods
- Not supported or ignored Telnet negotiation options
- Different error messages returned
- and more…

Results Telnet – Internet wide scans (1/3)

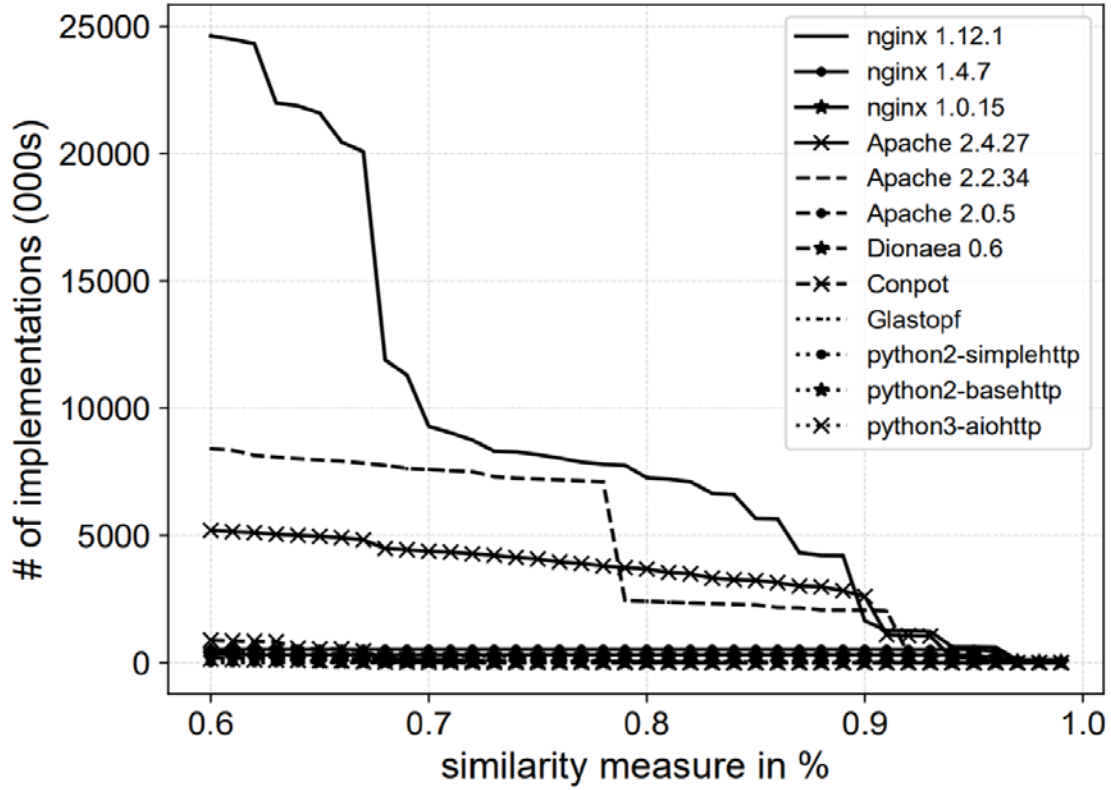
- First study to give an estimate of Telnet implementations
- Most implementations are similar to Busybox 1.6-2.4
- Not many servers respond in the same way as honeypots



Results SSH/HTTP – Internet wide scans (2/3)



Most implementations are similar to OpenSSH 6.6 and OpenSSH 7.2



Most implementations are similar to nginx 1.12.1, Apache 2.2.34 and Apache 2.4.27

Results Honeypots – Internet wide scans (3/3)

	Date	#ACKs	Sum	Kippo	Cowrie
Scan 1 (SSH)	2017-09	18,196k	2844	906	1938
Scan 2 (SSH)	2018-01	20,586k	2779	758	2021

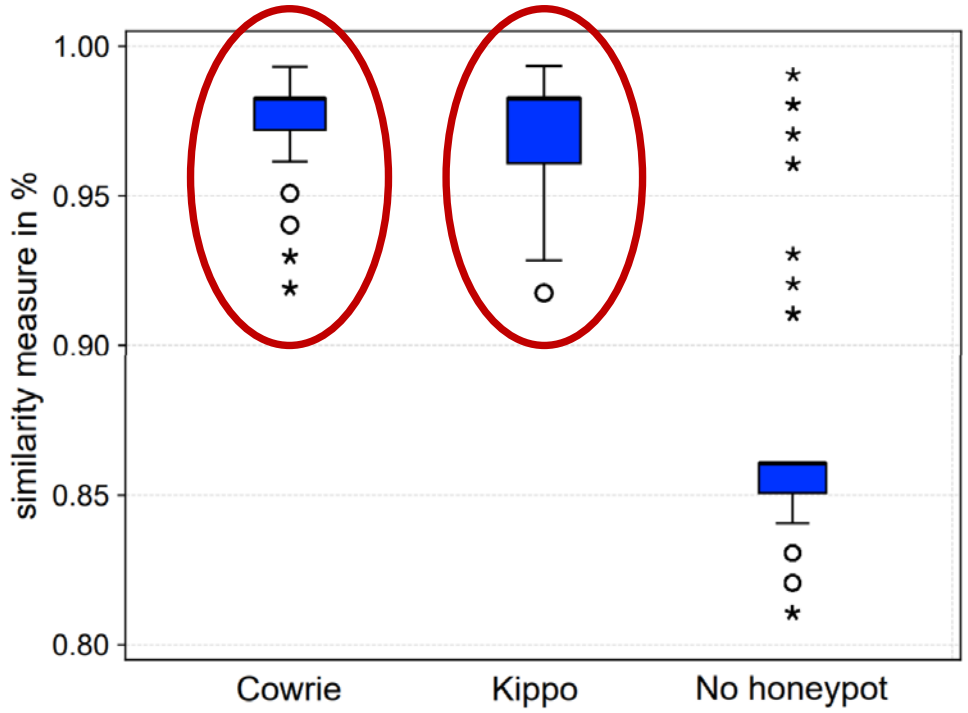
				TPwd	MTPot	TIoT	Cowrie
Scan 1 (Telnet)	2017-09	8,290k	1430	1	388	22	1019
Scan 2 (Telnet)	2018-01	8,169k	1166	1	216	11	938

				Dionaea	Glastopf	Conpot
Scan 1 (HTTP)	2017-10	58,775k	2616	139	2390	87
Scan 2 (HTTP)	2018-01	67,615k	3660	202	3371	87

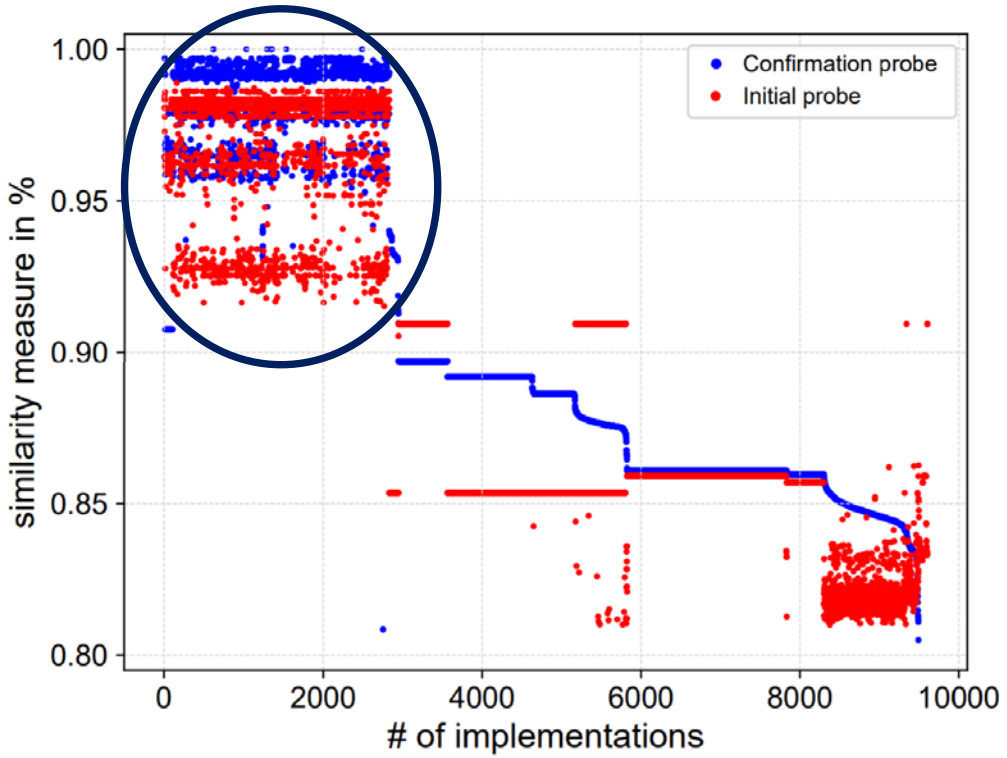
Validation and Accuracy (1/2)

Random padding of packets does not allow for exact matches

Removing the random parts



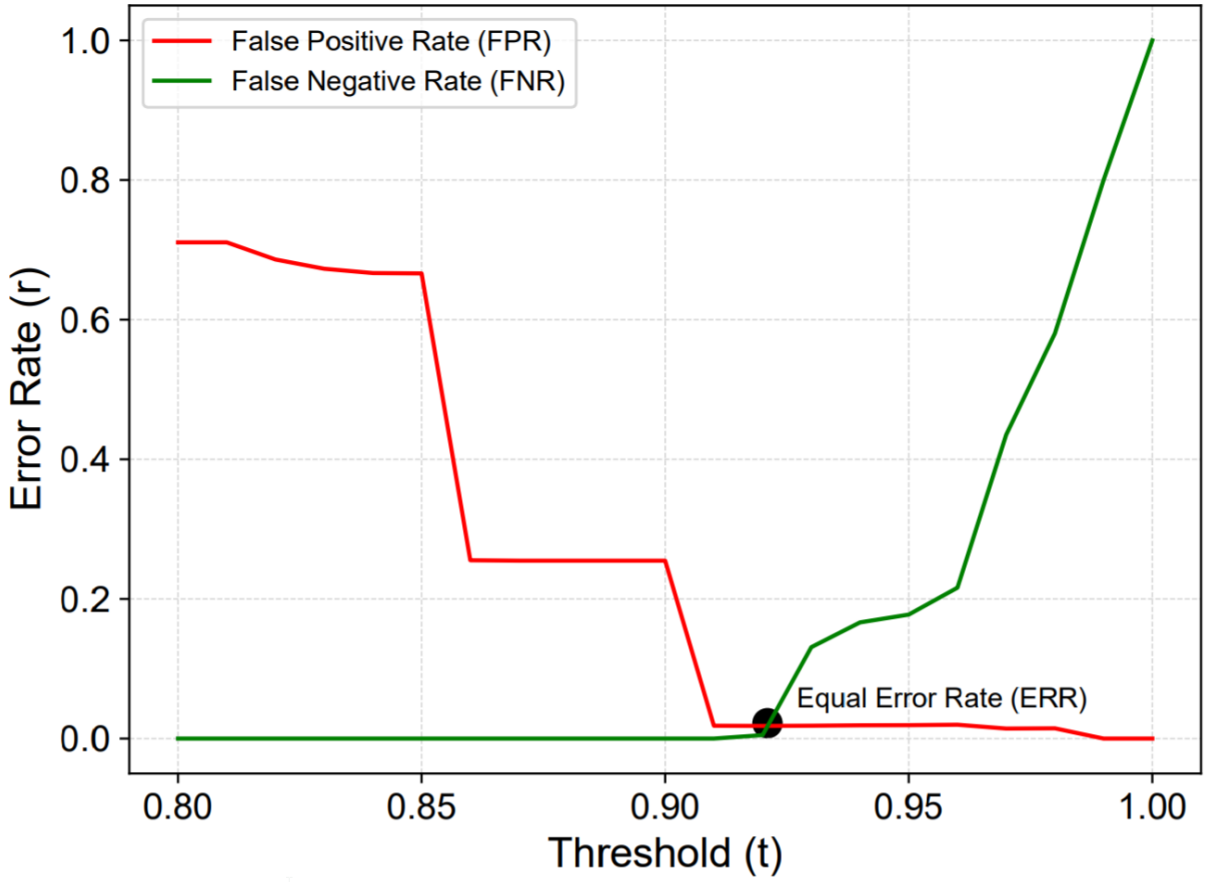
Use second-best distinguishing probe



Validation and Accuracy (2/2)

Equal Error Rate (ERR) of 0.0183

- We falsely accept and at the same time fail to identify 51 honeypots
- 2,779 honeypots as ‘ground truth’



Results – Mass Deployment

- 724 IPs run both an SSH and Web honeypot
- Many honeypots are hosted at well-known cloud providers

CO	ASN	Organisation	Telnet	SSH	HTTP	Total
US	16509	Amazon.com	140	520	506	1166
JP	2500	WIDE Project	–	–	490	490
US	14061	Digital Ocean	162	189	139	490
FR	16276	OVH SAS	117	202	122	441
TW	4662	GCNet	15	2	254	271
TW	18182	Sony Network	2	–	256	258
US	15169	Google LLC	45	139	46	230
TW	9924	Taiwan Fixed	1	74	146	221
US	14618	Amazon.com	12	70	110	192
RO	43443	DDNET Sol.	30	–	155	185

Results (SSH) – Configuration

- Only 79% of SSH honeypots have an unique host key
- SSH Honeypot operators rarely update their honeypots

		Scan 1 (SSH)		Scan 2 (SSH)	
Kippo	<2014-05-28	695	(24.4%)	546	(19.6%)
Kippo	<2015-05-24	211	(7.4%)	212	(7.6%)
Cowrie	<2017-06-06	1228	(43.2%)	950	(34.2%)
Cowrie	≤date of scan	710	(25.0%)	1071	(38.6%)

Impact and Countermeasures

We can detect your honeypots without even trying to send any credentials

- It is hard to tell from the logging that you've been detected!
- It is easy to add scripts using these techniques into tools such as Metasploit!

Closely monitor and update your honeypots

- Honeypot operators are as bad as anyone with patching

Patching against the specific distinguishers we report in the paper is not a solution as there are thousands more

- We developed a modified version of the OpenSSH daemon (sshd) which can front-end a Cowrie instance so that the protocol layer distinguishers will no longer work

Ethical Considerations

- We followed our institution's ethical research policy
  - with appropriate authorisation at every stage
- We used the exclusion list maintained by DNS-OARC
- We notified all local CERTs of our scans
- We respected requests to be excluded from further scanning
- We notified the relevant honeypot and library developers of our findings

Conclusion

Presented a generic approach for fingerprinting honeypots (“class break”)

- With a TCP handshake and usually one further packet we identify if you are running Kippo, Cowrie, Glastopf or various other (we believe all) low- and medium-interaction honeypots

Performed Internet wide scans for 9 different honeypots

- Found 7,605 honeypots residing on 6,125 IPv4 addresses
- Majority are hosted at well known cloud providers
- Only 39% of SSH honeypots were updated within the previous 7 months

We need a new architecture for low- and medium-interaction honeypots

- The “bad guys” can easily reproduce and implement our techniques

Q & A

Alexander Vetterl

alexander.vetterl@cl.cam.ac.uk

https://github.com/amv42/sshd-honeypot