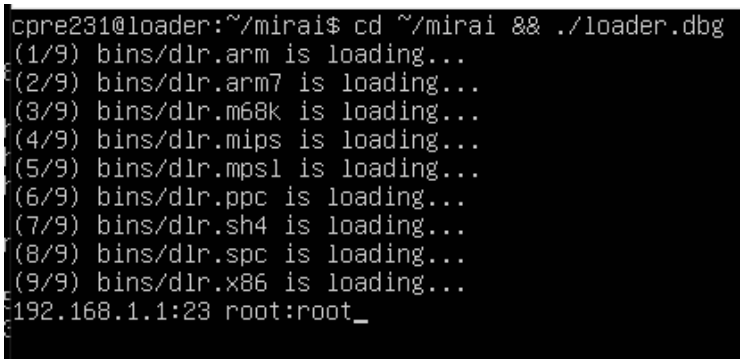
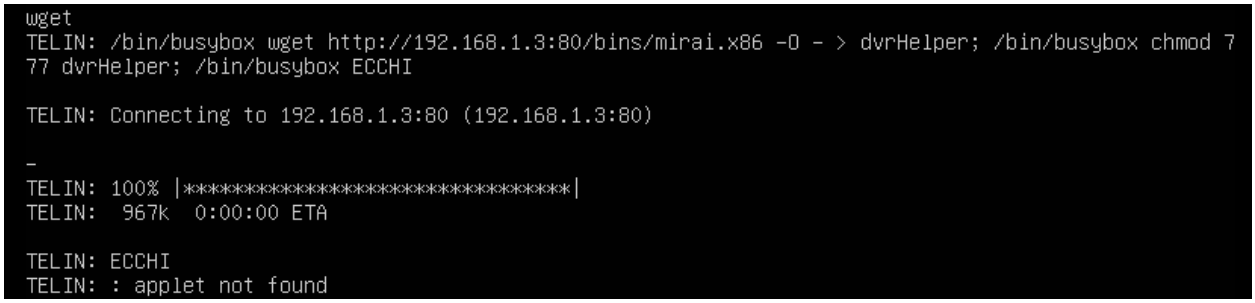
1. Identify two additional architectures for which the Mirai dropper has been compiled. Don't give me just the abbreviations. You must give me the full names. Just because a device has this architecture, doesn't mean it is susceptible to Mirai. What other feature would need to be enabled?



It is able to run on ARM (which I didn't know used to stand for either Advanced RISC Machines and originally Acorn RISC Machine) architectures and it can be used on the PowerPC architecture. However, the device must use default username / passwords to login and control the system for Mirai to be installed. (It also seems like they have to run Linux because they use busybox.)

2. Explain the line containing three commands.



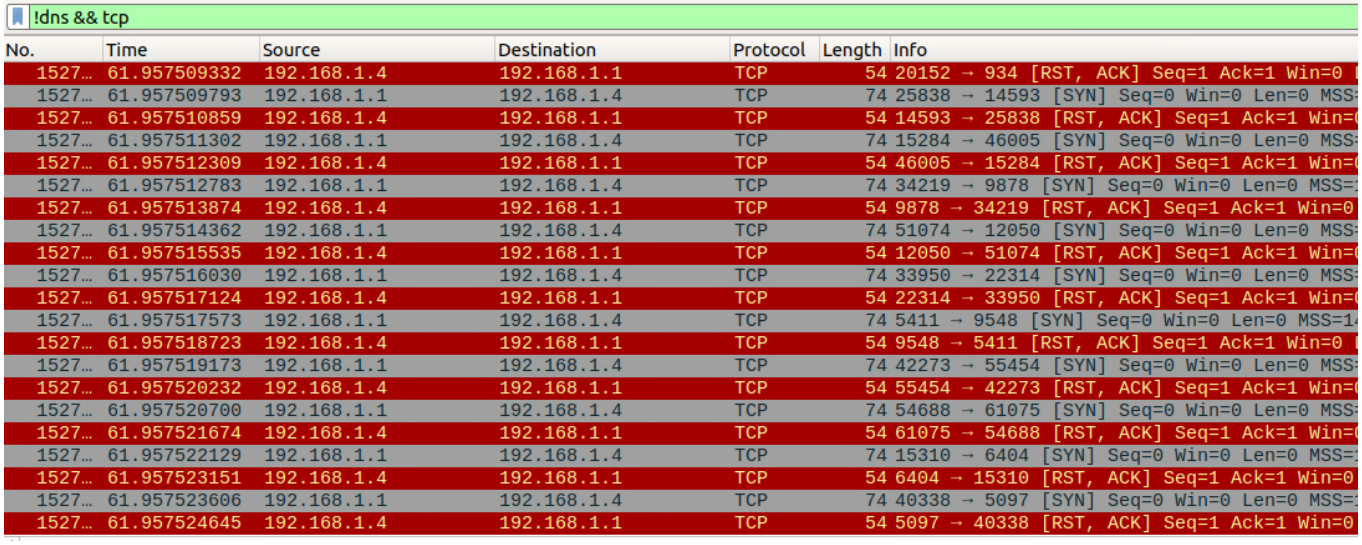
The first command uses wget to download the mirai.x86 code and stores it within a file called dvrHelper. dvrHelper is then made to be read, write, and executable to anyone (most importantly executable). It then runs something called ECCHI, which says it is not found. It uses Busybox to run the commands.

3. What are the two functions of the dvrHelper file?
I'm not exactly sure but it looks like it will establish a tunnel creating the tun0 device probably with the cnc. Based on the fact it takes telnet.x86 as a parameter, it probably then opens up and listens on telnet over that tunnel to wait for commands becoming a bot.

4. Screenshot of help (?) from the CNC command line.

```
mirai@botnet# ?
Available attack list
dns: DNS resolver flood using the targets domain, input IP is ignored
syn: SYN flood
ack: ACK flood
stomp: TCP stomp flood
greeth: GRE Ethernet flood
udpplain: UDP flood with less options. optimized for higher PPS
vse: Valve source engine specific flood
greip: GRE IP flood
http: HTTP flood
udp: UDP flood
```

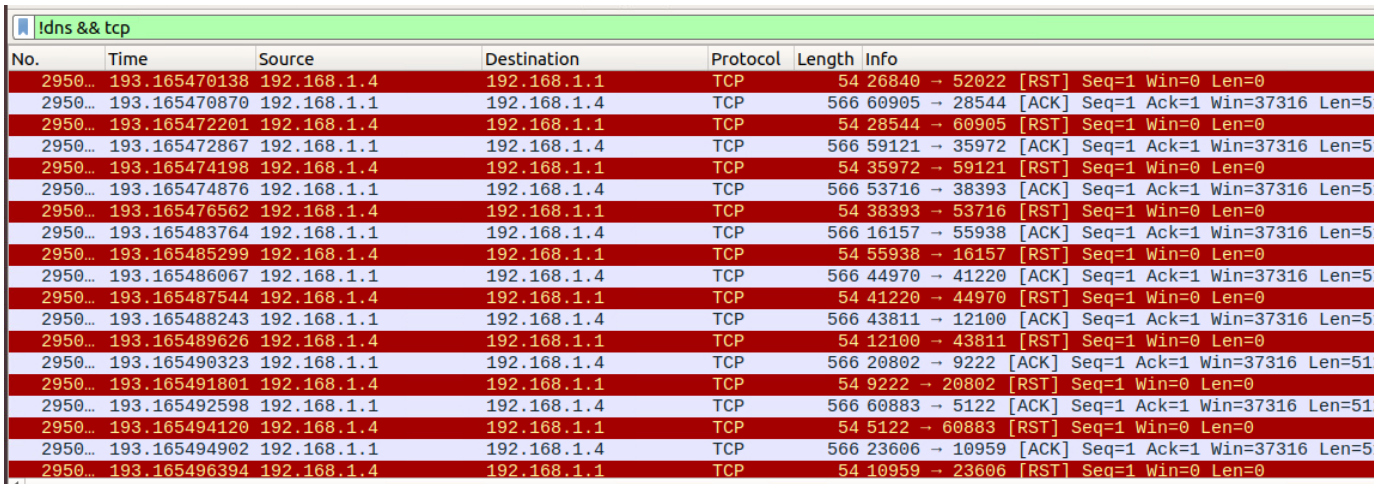
5. Screenshot of wireshark of successful syn flood attack



!dns && tcp

No.	Time	Source	Destination	Protocol	Length	Info
1527…	61.957509332	192.168.1.4	192.168.1.1	TCP	54	20152 → 934 [RST, ACK] Seq=1 Ack=1 Win=0
1527…	61.957509793	192.168.1.1	192.168.1.4	TCP	74	25838 → 14593 [SYN] Seq=0 Win=0 Len=0 MSS=
1527…	61.957510859	192.168.1.4	192.168.1.1	TCP	54	14593 → 25838 [RST, ACK] Seq=1 Ack=1 Win=0
1527…	61.957511302	192.168.1.1	192.168.1.4	TCP	74	15284 → 46005 [SYN] Seq=0 Win=0 Len=0 MSS=
1527…	61.957512309	192.168.1.4	192.168.1.1	TCP	54	46005 → 15284 [RST, ACK] Seq=1 Ack=1 Win=0
1527…	61.957512783	192.168.1.1	192.168.1.4	TCP	74	34219 → 9878 [SYN] Seq=0 Win=0 Len=0 MSS=
1527…	61.957513874	192.168.1.4	192.168.1.1	TCP	54	9878 → 34219 [RST, ACK] Seq=1 Ack=1 Win=0
1527…	61.957514362	192.168.1.1	192.168.1.4	TCP	74	51074 → 12050 [SYN] Seq=0 Win=0 Len=0 MSS=
1527…	61.957515535	192.168.1.4	192.168.1.1	TCP	54	12050 → 51074 [RST, ACK] Seq=1 Ack=1 Win=0
1527…	61.957516030	192.168.1.1	192.168.1.4	TCP	74	33950 → 22314 [SYN] Seq=0 Win=0 Len=0 MSS=
1527…	61.957517124	192.168.1.4	192.168.1.1	TCP	54	22314 → 33950 [RST, ACK] Seq=1 Ack=1 Win=0
1527…	61.957517573	192.168.1.1	192.168.1.4	TCP	74	5411 → 9548 [SYN] Seq=0 Win=0 Len=0 MSS=14
1527…	61.957518723	192.168.1.4	192.168.1.1	TCP	54	9548 → 5411 [RST, ACK] Seq=1 Ack=1 Win=0
1527…	61.957519173	192.168.1.1	192.168.1.4	TCP	74	42273 → 55454 [SYN] Seq=0 Win=0 Len=0 MSS=
1527…	61.957520232	192.168.1.4	192.168.1.1	TCP	54	55454 → 42273 [RST, ACK] Seq=1 Ack=1 Win=0
1527…	61.957520700	192.168.1.1	192.168.1.4	TCP	74	54688 → 61075 [SYN] Seq=0 Win=0 Len=0 MSS=
1527…	61.957521674	192.168.1.4	192.168.1.1	TCP	54	61075 → 54688 [RST, ACK] Seq=1 Ack=1 Win=0
1527…	61.957522129	192.168.1.1	192.168.1.4	TCP	74	15310 → 6404 [SYN] Seq=0 Win=0 Len=0 MSS=
1527…	61.957523151	192.168.1.4	192.168.1.1	TCP	54	6404 → 15310 [RST, ACK] Seq=1 Ack=1 Win=0
1527…	61.957523606	192.168.1.1	192.168.1.4	TCP	74	40338 → 5097 [SYN] Seq=0 Win=0 Len=0 MSS=
1527…	61.957524645	192.168.1.4	192.168.1.1	TCP	54	5097 → 40338 [RST, ACK] Seq=1 Ack=1 Win=0

6. Screenshot of wireshark of any other successful attack from Mirai's list of options
   Ack flood



!dns && tcp

No.	Time	Source	Destination	Protocol	Length	Info
2950…	193.165470138	192.168.1.4	192.168.1.1	TCP	54	26840 → 52022 [RST] Seq=1 Win=0 Len=0
2950…	193.165470870	192.168.1.1	192.168.1.4	TCP	566	60905 → 28544 [ACK] Seq=1 Ack=1 Win=37316 Len=5
2950…	193.165472201	192.168.1.4	192.168.1.1	TCP	54	28544 → 60905 [RST] Seq=1 Win=0 Len=0
2950…	193.165472867	192.168.1.1	192.168.1.4	TCP	566	59121 → 35972 [ACK] Seq=1 Ack=1 Win=37316 Len=5
2950…	193.165474198	192.168.1.4	192.168.1.1	TCP	54	35972 → 59121 [RST] Seq=1 Win=0 Len=0
2950…	193.165474876	192.168.1.1	192.168.1.4	TCP	566	53716 → 38393 [ACK] Seq=1 Ack=1 Win=37316 Len=5
2950…	193.165476562	192.168.1.4	192.168.1.1	TCP	54	38393 → 53716 [RST] Seq=1 Win=0 Len=0
2950…	193.165483764	192.168.1.1	192.168.1.4	TCP	566	16157 → 55938 [ACK] Seq=1 Ack=1 Win=37316 Len=5
2950…	193.165485299	192.168.1.4	192.168.1.1	TCP	54	55938 → 16157 [RST] Seq=1 Win=0 Len=0
2950…	193.165486067	192.168.1.1	192.168.1.4	TCP	566	44970 → 41220 [ACK] Seq=1 Ack=1 Win=37316 Len=5
2950…	193.165487544	192.168.1.4	192.168.1.1	TCP	54	41220 → 44970 [RST] Seq=1 Win=0 Len=0
2950…	193.165488243	192.168.1.1	192.168.1.4	TCP	566	43811 → 12100 [ACK] Seq=1 Ack=1 Win=37316 Len=5
2950…	193.165489626	192.168.1.4	192.168.1.1	TCP	54	12100 → 43811 [RST] Seq=1 Win=0 Len=0
2950…	193.165490323	192.168.1.1	192.168.1.4	TCP	566	20802 → 9222 [ACK] Seq=1 Ack=1 Win=37316 Len=51
2950…	193.165491801	192.168.1.4	192.168.1.1	TCP	54	9222 → 20802 [RST] Seq=1 Win=0 Len=0
2950…	193.165492598	192.168.1.1	192.168.1.4	TCP	566	60883 → 5122 [ACK] Seq=1 Ack=1 Win=37316 Len=51
2950…	193.165494120	192.168.1.4	192.168.1.1	TCP	54	5122 → 60883 [RST] Seq=1 Win=0 Len=0
2950…	193.165494902	192.168.1.1	192.168.1.4	TCP	566	23606 → 10959 [ACK] Seq=1 Ack=1 Win=37316 Len=5
2950…	193.165496394	192.168.1.4	192.168.1.1	TCP	54	10959 → 23606 [RST] Seq=1 Win=0 Len=0

7. What were some of the devices first used in Mirai attacks? Please include the manufacturer and the model. What are some of the commonalities found in the devices in your previous answer?

   This is essentially the devices I could find when looking online. They are from:
   https://us-cert.cisa.gov/ics/alerts/ICS-ALERT-16-286-01
   https://krebsonsecurity.com/2016/10/who-makes-the-iot-things-under-attack/

   The major commonality that I see is that the devices are mostly cameras or routers. Which makes sense that they can be targeted as IoT cameras main staple is that you can view them from your phone from anywhere and routers handle your public IP so they are publicly accessible.

   This alert is being produced to amplify mitigations outlined by Sierra Wireless, for users of the following products:

   - LS300,
   - GX400,
   - GX/ES440,
   - GX/ES450, and
   - RV50

Username/Password	Manufacturer	Link to supporting evidence
admin/123456	ACTi IP Camera	https://ipvm.com/reports/ip-cameras-default-passwords-directory
root/anko	ANKO Products DVR	http://www.cctvforum.com/viewtopic.php?f=3&t=44250
root/pass	Axis IP Camera, et. al	http://www.cleancss.com/router-default/Axis/0543-001
root/vizxv	Dahua Camera	http://www.cam-it.org/index.php?topic=5192.0
root/888888	Dahua DVR	http://www.cam-it.org/index.php?topic=5035.0
root/666666	Dahua DVR	http://www.cam-it.org/index.php?topic=5035.0
root/7ujMko0vizxv	Dahua IP Camera	http://www.cam-it.org/index.php?topic=9396.0
root/7ujMko0admin	Dahua IP Camera	http://www.cam-it.org/index.php?topic=9396.0
666666/666666	Dahua IP Camera	http://www.cleancss.com/router-default/Dahua/DH-IPC-HDW4300C
root/dreambox	Dreambox TV receiver	https://www.satellites.co.uk/forums/threads/reset-root-password-plugin.101146/
root/zlxx	EV ZLX Two-way Speaker?	?
root/juantech	Guangzhou Juan Optical	https://news.ycombinator.com/item?id=11114012
root/xc3511	H.264 - Chinese DVR	http://www.cctvforum.com/viewtopic.php?f=56&t=34930&start=15
root/hi3518	HiSilicon IP Camera	https://acassis.wordpress.com/2014/08/10/i-got-a-new-hi3518-ip-camera-modules/
root/klv123	HiSilicon IP Camera	https://gist.github.com/gabonator/74cdd6ab4f733ff047356198c781f27d
root/klv1234	HiSilicon IP Camera	https://gist.github.com/gabonator/74cdd6ab4f733ff047356198c781f27d
root/jvbzd	HiSilicon IP Camera	https://gist.github.com/gabonator/74cdd6ab4f733ff047356198c781f27d
root/admin	IPX-DDK Network Camera	http://www.ipxinc.com/products/cameras-and-video-servers/network-cameras/
root/system	IQinVision Cameras, et. al	https://ipvm.com/reports/ip-cameras-default-passwords-directory
admin/meinsm	Mobotix Network Camera	http://www.forum.use-ip.co.uk/threads/mobotix-default-password.76/
root/54321	Packet8 VOIP Phone, et. al	http://webcache.googleusercontent.com/search?q=cache:W1phozQZURUJ:community.freepbx.org/t/packet8-atas-phones/411
root/00000000	Panasonic Printer	https://www.experts-exchange.com/questions/26194395/Default-User-Password-for-Panasonic-DP-C405-Web-Interface.html
root/realtek	RealTek Routers	
admin/1111111	Samsung IP Camera	https://ipvm.com/reports/ip-cameras-default-passwords-directory
root/xmhdipc	Shenzhen Anran Security Camera	https://www.amazon.com/MegaPixel-Wireless-Network-Surveillance-Camera/product-reviews/B00EB6FNDI
admin/smcadmin	SMC Routers	http://www.cleancss.com/router-default/SMC/ROUTER
root/ikwb	Toshiba Network Camera	http://faq.surveillixdvrsupport.com/index.php?action=artikel&cat=4&id=8&artlang=en
ubnt/ubnt	Ubiquiti AirOS Router	http://setuprouter.com/router/ubiquiti/airos-airgrid-m5hp/login.htm
supervisor/supervisor	VideoIQ	https://ipvm.com/reports/ip-cameras-default-passwords-directory
root/<none>	Vivotek IP Camera	https://ipvm.com/reports/ip-cameras-default-passwords-directory
admin/1111	Xerox printers, et. al	https://atyourservice.blogs.xerox.com/2012/08/28/logging-in-as-system-administrator-on-your-xerox-printer/
root/Zte521	ZTE Router	http://www.ironbugs.com/2016/02/hack-and-patch-your-zte-f660-routers.html

8. Why did Mirai not try to attack private IP addresses, the USPS, nor the DoD?
   It probably tries not to infect private IP addresses as they wouldn't publicly addressable for the servers to communicate with. The other IPs are probably since they are high ticket agencies that could and would stop at nothing to track down the individual especially if they were attacked.