



# Open Source Threat Intelligence Chat Bot

**Tony Lee** | Sr. Technical Director  
(Chief Minion Herder)

# TONY@CYLANCE: ~ \$ WHOAMI



## ▪ 14 Years of Professional Security Experience

- Currently Sr. Technical Director at Cylance, Inc.

## ▪ Specialties

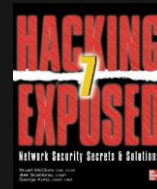
- Offensive security / Red teaming
- Rapid prototyping and product integration

## ▪ Education

- Bachelor's in Computer Engineering, Virginia Tech
- Master's in Security Informatics, Johns Hopkins University

## ▪ Research and Publications

- Contributing author to Hacking Exposed 7 and frequent blogger
- Wireless security, China Chopper web shell, Cisco's SYNful Knock router implant
- Forensic Investigator Splunk app



# AGENDA

- Introduction
- Components
  - Software
  - Hardware
- Installation & Example Configuration
- Usage & Features
- Development
- Acknowledgements
- Demo Setup

# INTRODUCTION

- Are you a SOC analyst, incident responder, or threat researcher?
- Do you get tired of pivoting to multiple tools to perform your investigation?
- Have you ever wanted a minion to do your research and bidding, but have limited time and budget?

Let **CyBot** be your minion!



# COMPONENTS

## ▪ Software

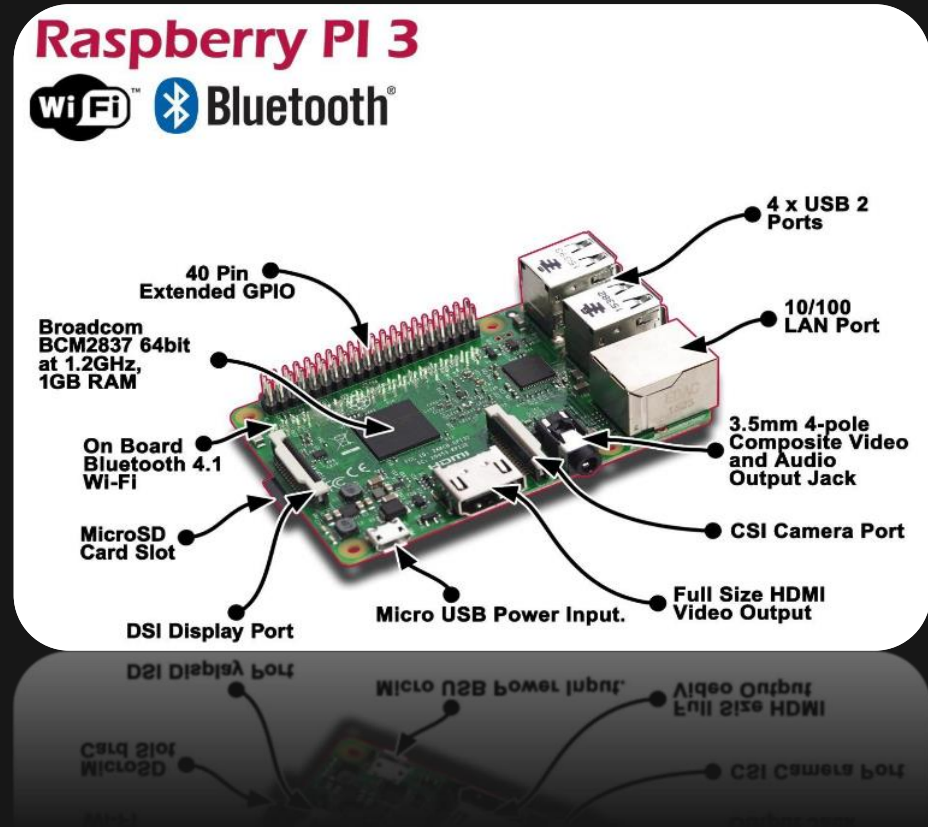
- Linux, Mac, Windows
- Python3
- ErrBot
- CyBot Plugins from GitHub



## ▪ Hardware

- VM or Unused hardware that is laying around
- or-
- Raspberry Pi
- Example:

[https://www.amazon.com/gp/product/B01CUMNIV8/ref=oh\\_aui\\_detailpage\\_o01\\_s00?ie=UTF8&pssc=1](https://www.amazon.com/gp/product/B01CUMNIV8/ref=oh_aui_detailpage_o01_s00?ie=UTF8&pssc=1)



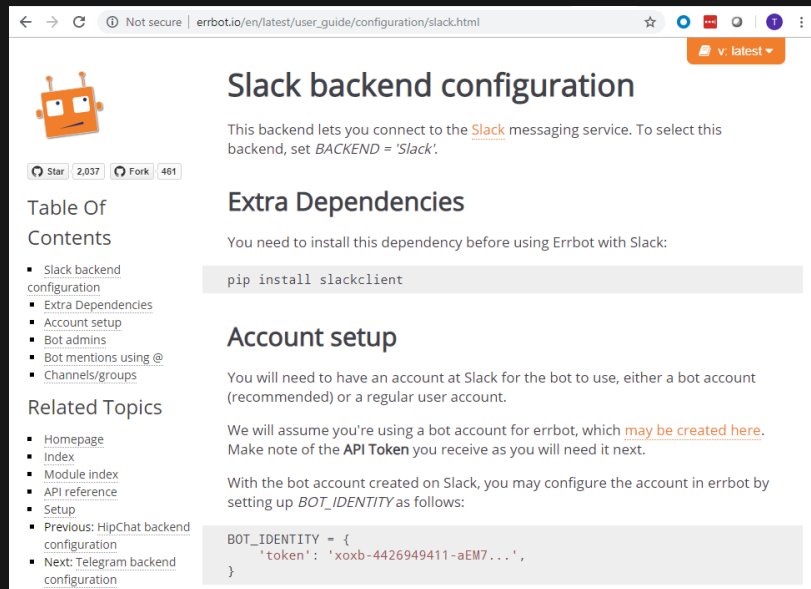
# COMPONENTS

- **ErrBot is very flexible and supports multiple server backends**

- XMPP (Any standards-compliant XMPP/Jabber server should work - Google Talk/Hangouts included)
- IRC
- **Slack**
- Telegram
- Tox (maintained separately)
- Gitter (maintained separately)
- CampFire (maintained separately)
- Skype (maintained separately)
- Zulip (maintained separately)

- **Configuration largely depends on the chat program**

- Our example configuration will use Slack!
- Source: [http://errbot.io/en/latest/user\\_guide/configuration/slack.html](http://errbot.io/en/latest/user_guide/configuration/slack.html)



The screenshot shows a web browser window with the URL `errbot.io/en/latest/user_guide/configuration/slack.html`. The page title is "Slack backend configuration". It features the ErrBot logo (an orange robot head) and GitHub statistics (2,037 stars, 461 forks). The page is divided into several sections: "Table Of Contents" with links to "Slack backend configuration", "Extra Dependencies", "Account setup", "Bot admins", "Bot mentions using @", and "Channels/groups"; "Extra Dependencies" with the command `pip install slackclient`; "Account setup" with instructions on how to create a bot account and obtain an API token; and "Related Topics" with links to "Homepage", "Index", "Module index", "API reference", "Setup", "Previous: HipChat backend configuration", and "Next: Telegram backend configuration". A code block at the bottom shows the `BOT_IDENTITY` configuration dictionary.

```
BOT_IDENTITY = {
    'token': 'xoxb-4426949411-aEM7...',
}
```

# INSTALLATION

## ▪ Prerequisites

```
sudo apt-get install python3 python-dev libssl-dev python3-pip
```

```
sudo pip3 install errbot
```

```
mkdir ~/errbot-root
```

```
cd ~/errbot-root
```

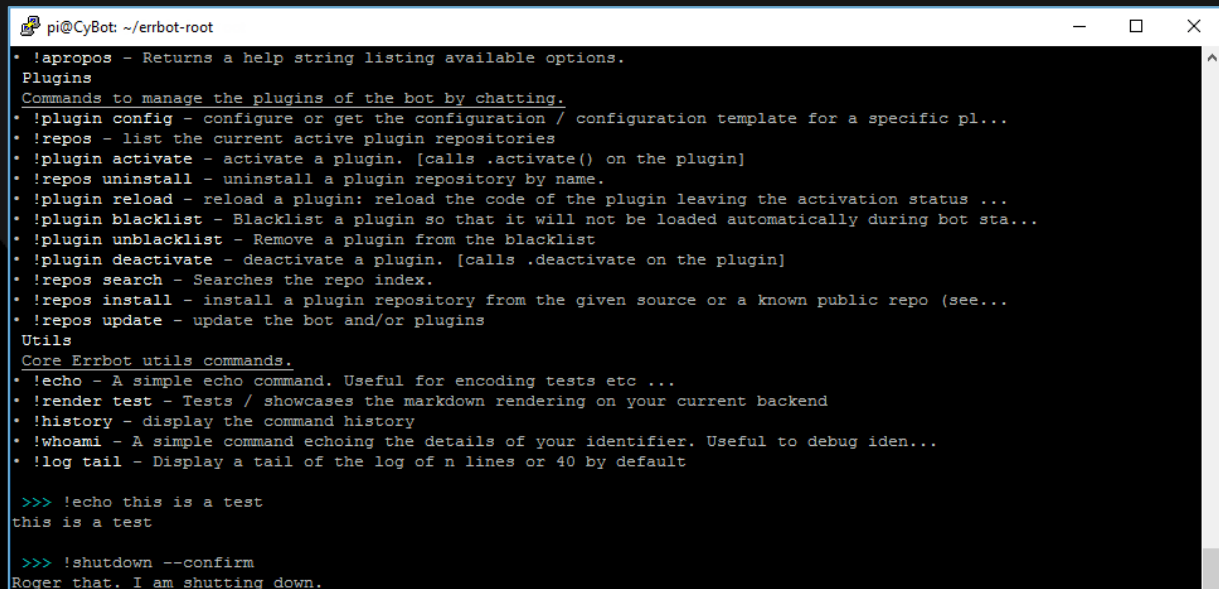
```
errbot --init
```

## ▪ Validation

```
errbot
```

```
!help
```

```
!shutdown --confirm
```



```
pi@CyBot: ~/errbot-root
* !apropos - Returns a help string listing available options.
Plugins
Commands to manage the plugins of the bot by chatting.
* !plugin config - configure or get the configuration / configuration template for a specific pl...
* !repos - list the current active plugin repositories
* !plugin activate - activate a plugin. [calls .activate() on the plugin]
* !repos uninstall - uninstall a plugin repository by name.
* !plugin reload - reload a plugin: reload the code of the plugin leaving the activation status ...
* !plugin blacklist - Blacklist a plugin so that it will not be loaded automatically during bot sta...
* !plugin unblacklist - Remove a plugin from the blacklist
* !plugin deactivate - deactivate a plugin. [calls .deactivate on the plugin]
* !repos search - Searches the repo index.
* !repos install - install a plugin repository from the given source or a known public repo (see...
* !repos update - update the bot and/or plugins
Utils
Core Errbot utils commands.
* !echo - A simple echo command. Useful for encoding tests etc ...
* !render test - Tests / showcases the markdown rendering on your current backend
* !history - display the command history
* !whoami - A simple command echoing the details of your identifier. Useful to debug iden...
* !log tail - Display a tail of the log of n lines or 40 by default

>>> !echo this is a test
this is a test

>>> !shutdown --confirm
Roger that. I am shutting down.
```

# EXAMPLE CONFIGURATION

- Install ErrBot Slack components

```
sudo pip3 install slack client
```

- Setup bot account in Slack Web UI
  - Note: You must be a Workspace Owner who has permission to add integrations
  - Go here: <https://my.slack.com/services/new/bot>
  - Create the bot account and take note of the API Token



# EXAMPLE CONFIGURATION

- Necessities for config.py

```
BACKEND = 'Slack' # defaults to XMPP
BOT_DATA_DIR = r' /<path>/errbot-root/data'
BOT_EXTRA_PLUGIN_DIR = ' /<path>/errbot-root/plugins'
BOT_LOG_FILE = r' /<path>/errbot-root/errbot.log'
BOT_LOG_LEVEL = logging.INFO

BOT_IDENTITY = {
    'token' : 'ed4b74d628example312ff04',
}

CHATROOM_PRESENCE = ()
CHATROOM_FN = 'CyBot'
```

# EXAMPLE CONFIGURATION

- Added security for config.py

```
BOT_ADMINS = ('@Tony',)

ACCESS_CONTROLS = {'status': {'allowrooms': ('someroom@conference.localhost',)},
                   'private_plugin': {'allowprivate': ('#protected_room',)},
                   'about': {'denyusers': ('*@evilhost',), 'allowrooms':
('room1@conference.localhost', 'room2@conference.localhost')},
                   'uptime': {'allowusers': BOT_ADMINS},
                   'help': {'allowmuc': False},
                   'help': {'allowmuc': False},
                   'ChatRoom:*': {'allowusers': BOT_ADMINS},
                   }

DIVERT_TO_PRIVATE = ('help', 'about', 'status', 'secnews', 'vulnnews', 'ransom', 'threat',
'aptgroup', 'cve', 'hacktool')
```

# USAGE AND FEATURES

- **Control CyBot (ErrBot specific commands)**

Command	Arguments	Description
!help	N/A	Lists the help page
!room-list	N/A	Lists the rooms that the bot is monitoring
!room-join	XMPP room name	Joins a room to monitor
!room-destroy	XMPP room name	Deletes a room — good for accidental creation
!restart	N/A	Useful for reloading plugins

- **Notes:**

- We recommend allowing CyBot to respond to private messages in a controlled chat environment
- CyBot must be a member of the room to respond to commands within a chat room
- Bot accounts on Slack are not allowed to join/leave channels on their own (they must be invited by a user instead)

# USAGE AND FEATURES

## ▪ Current File and Network Commands

Command	Arguments	Description
!vt	<hash   URL>	VirusTotal Query
!hashid	<hash>	Identifies a hash type
!safebrowsing	<URL>	Google Safebrowsing Lookup
!whois	<domain or IP>	WHOIS Query
!nslookup	<domain or IP>	Forward and Reverse DNS Lookups
!geoip	<FQDN   IP>	Perform GeoIP lookup of hosts
!unshorten	<Shortened URL>	Unshorten URLs such as goo.gl and more
!linkextractor	<FQDN   IP>	Extracts links from a site
!urldecode	<URL>	Decodes an encoded URL
!uastring	<UA string in quotes>	Convert Unix time to human readable

# USAGE AND FEATURES

## ▀ Current Threat and Vulnerability Research Commands

Command	Arguments	Description
!ransom	<search string>	Identify ransomware by searching the Ransomware Overview Spreadsheet
!threat	<search string>	Search APT group activity mapped to MITRE ATT&CK Framework
!aptgroup	<search string>	Retrieve information on common APT groups
!hacktool	<search string>	Retrieve information on common hacking tools
!cve	<#   #####-####>	Return the last n CVE's or specify #####-#### CVE
!secnews	N/A	Displays latest cyber security news
!vulnnews	N/A	Displays latest computer vulnerability news

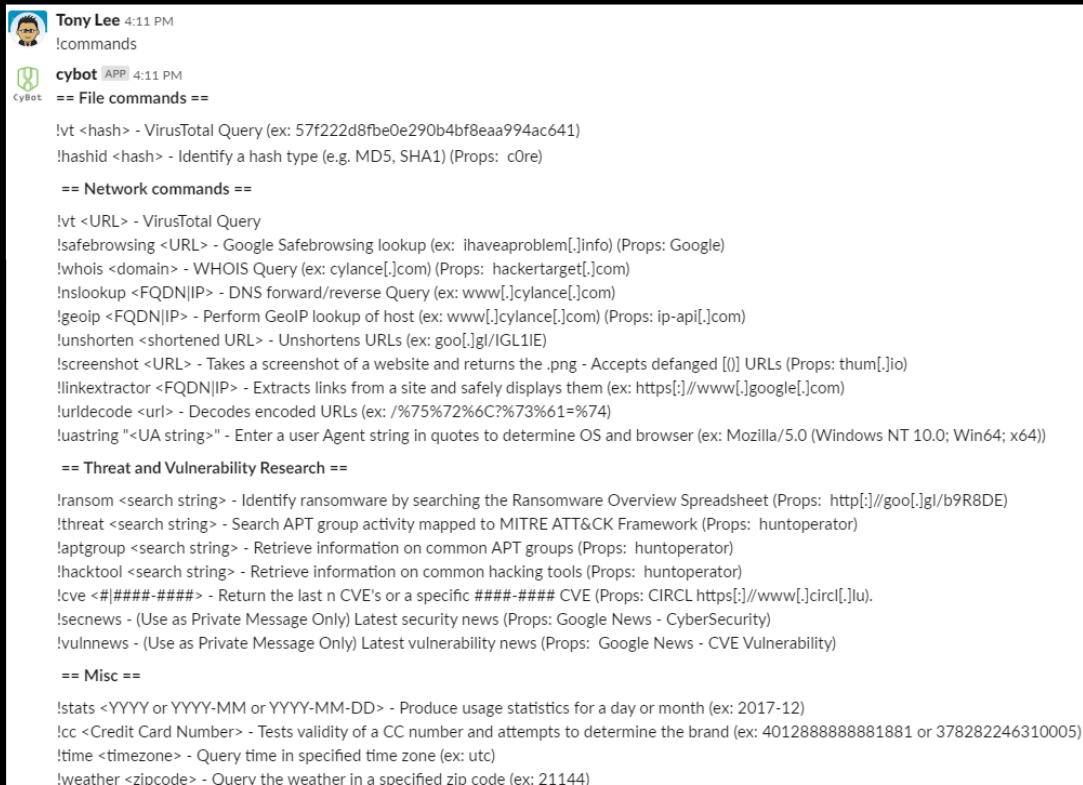
# USAGE AND FEATURES

## ▪ Current Misc Commands

Command	Arguments	Description
!stats	<YYYY or YYYY-MM or YYYY-MM-DD>	Produce usage statistics for a day or month or year (ex: 2017-12)
!cc	<Credit Card Number>	Tests validity of a CC number and attempt to determine the brand
!time	<timezone>	Query time in specified timezone
!weather	<zipcode>	Query the weather in a specified zip code
!unixtime	<epoch>	Convert Unix time (seconds since Jan 1, 1970) to human readable
!wintime	<epoch>	Convert Windows time (100-nanosecond intervals since January 1, 1601) to human readable (ex: 131580340430000000)
!calc	<arithmetic input>	Performs basic arithmetic (Valid input: [0-9]+-/ ) (ex: 22*3)
!bitcoin	N/A	Polls the latest Bitcoin Price Index (BPI)
!whatsyourip	N/A	Returns CyBot's public IP Address
!joke	N/A	Queries an on-line API repository of jokes
!codename	N/A	Generates a two word project codename

# USAGE AND FEATURES

## ▪ Example CyBot !commands menu



The screenshot shows a Telegram chat interface. At the top, a user named 'Tony Lee' with a profile picture of a person with glasses and a yellow background has sent a message at 4:11 PM: '!commands'. Below this, a bot named 'cybot' with a green shield icon and 'APP' label has responded at 4:11 PM with the text '== File commands =='. The bot's response lists various commands and their functions, categorized into File commands, Network commands, Threat and Vulnerability Research, and Misc. The commands include !vt, !hashid, !nslookup, !geoiip, !unshorten, !screenshot, !linkextractor, !urldecode, !uasttring, !ransom, !threat, !aptgroup, !hacktool, !cve, !secnews, !vulnnews, !stats, !cc, !time, and !weather.

4:11 PM  
!commands

cybot APP 4:11 PM  
== File commands ==

!vt <hash> - VirusTotal Query (ex: 57f222d8fbe0e290b4bf8eaa994ac641)  
!hashid <hash> - Identify a hash type (e.g. MD5, SHA1) (Props: c0re)

== Network commands ==

!vt <URL> - VirusTotal Query  
!safebrowsing <URL> - Google Safebrowsing lookup (ex: ihaveaproblem[.]info) (Props: Google)  
!whois <domain> - WHOIS Query (ex: cylance[.]com) (Props: hackertarget[.]com)  
!nslookup <FQDN|IP> - DNS forward/reverse Query (ex: www[.]cylance[.]com)  
!geoiip <FQDN|IP> - Perform GeolP lookup of host (ex: www[.]cylance[.]com) (Props: ip-api[.]com)  
!unshorten <shortened URL> - Unshortens URLs (ex: goo[.]gl/IGL1IE)  
!screenshot <URL> - Takes a screenshot of a website and returns the .png - Accepts defanged [0] URLs (Props: thum[.]jio)  
!linkextractor <FQDN|IP> - Extracts links from a site and safely displays them (ex: https[:]//www[.]google[.]com)  
!urldecode <url> - Decodes encoded URLs (ex: /%75%72%6C:%73%61=%74)  
!uasttring "<UA string>" - Enter a user Agent string in quotes to determine OS and browser (ex: Mozilla/5.0 (Windows NT 10.0; Win64; x64))

== Threat and Vulnerability Research ==

!ransom <search string> - Identify ransomware by searching the Ransomware Overview Spreadsheet (Props: http[:]//goo[.]gl/b9R8DE)  
!threat <search string> - Search APT group activity mapped to MITRE ATT&CK Framework (Props: huntoperator)  
!aptgroup <search string> - Retrieve information on common APT groups (Props: huntoperator)  
!hacktool <search string> - Retrieve information on common hacking tools (Props: huntoperator)  
!cve <#|####-####> - Return the last n CVE's or a specific ####-#### CVE (Props: CIRCL https[:]//www[.]circl[.]lu).  
!secnews - (Use as Private Message Only) Latest security news (Props: Google News - CyberSecurity)  
!vulnnews - (Use as Private Message Only) Latest vulnerability news (Props: Google News - CVE Vulnerability)

== Misc ==

!stats <YYYY or YYYY-MM or YYYY-MM-DD> - Produce usage statistics for a day or month (ex: 2017-12)  
!cc <Credit Card Number> - Tests validity of a CC number and attempts to determine the brand (ex: 4012888888881881 or 378282246310005)  
!time <timezone> - Query time in specified time zone (ex: utc)  
!weather <zipcode> - Query the weather in a specified zip code (ex: 21144)

# RELEASING AT BLACK HAT USA 2019

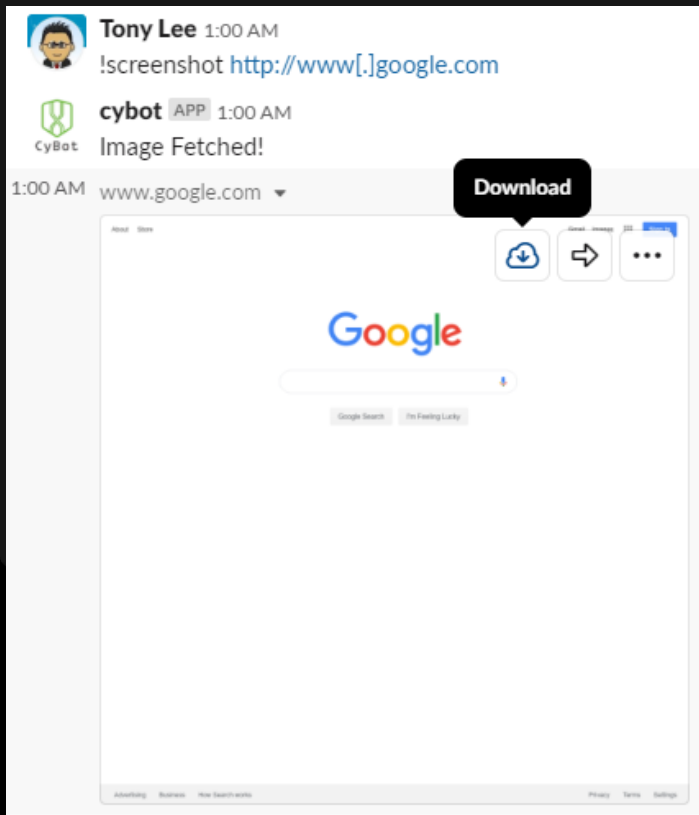
## ▪ New CyBot Plugin Features

Command	Arguments	Description
!screenshot	<Defanged_URL>	Takes a screenshot of a website & returns .png - Accepts defanged [()] URLs (Props: <a href="#">thum.io</a> )
!vtdownload	<hash>	VirusTotal Sample Download (Extension: zi_, Password:infected)
!cuckoo -s	N/A	Obtain the status of the Cuckoo instance
!cuckoo -u	<Defanged_URL>	Runs a URL through Cuckoo
!cuckoo -vt	<hash>	Downloads a hash from VT, password protect zips it, and submits it to Cuckoo
!cuckoo -v	<task_id>	Check the status of a task in Cuckoo
!cuckoo -r	<task_id>	Check the report of a task in Cuckoo



# RELEASING AT BLACK HAT USA 2019

▪ !screenshot <Defanged\_URL>



# RELEASING AT BLACK HAT USA 2019

▀ !vtdownload <hash>



**Tony Lee** 12:58 AM

!vtdownload 57f222d8fbe0e290b4bf8eaa994ac641



CyBot

**cybot** APP 12:58 AM

Fetching file: 57f222d8fbe0e290b4bf8eaa994ac641

Received HTTP 200. VirusTotal has the file!

Zipping the file with a password of: infected

Sending file via direct message with an extension of .zi\_

12:58 AM Zip ▾

Download



57f222d8fbe0e290b4bf8eaa994a

275 kB — Click to download



# RELEASING AT BLACK HAT USA 2019

▪ **!cuckoo**



**Tony Lee** 7:10 PM

**!cuckoo**



CyBot

**cybot** APP 7:10 PM

Incorrect number of parameters. Syntax: `!cuckoo -<s | u <Defanged_URL> | vt <hash> | v <task_id> | r <task_id>>`

`!cuckoo -s` (provides Cuckoo status)

`!cuckoo -u <Defanged_URL>` (runs a URL - [.] and (.) accepted)

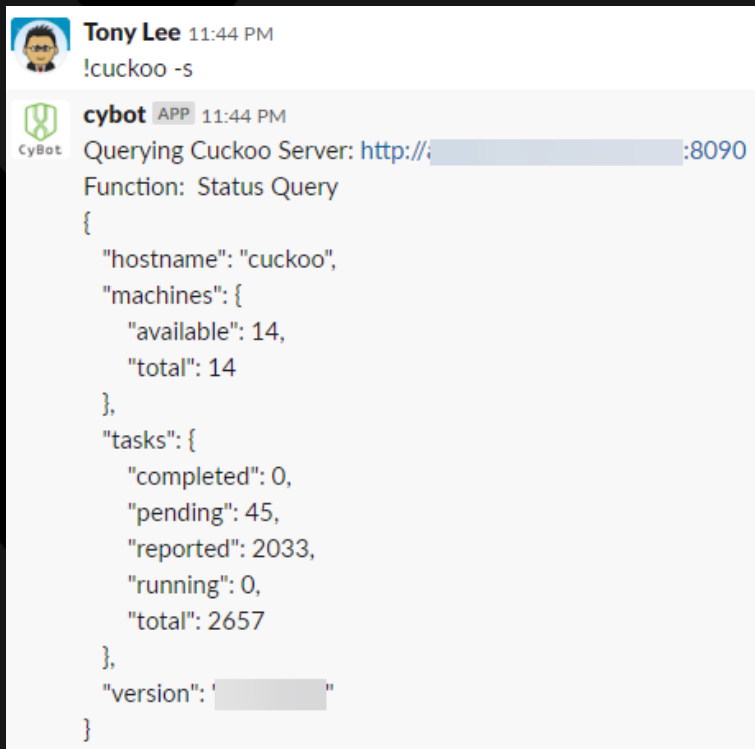
`!cuckoo -vt <hash>` (downloads a file from VT and runs the file)

`!cuckoo -v <task_id>` (views stats of task)

`!cuckoo -r <task_id>` (views report for task)

# RELEASING AT BLACK HAT USA 2019

▪ !cuckoo -s



# RELEASING AT BLACK HAT USA 2019

• `!cuckoo -u <defanged_URL>`



**Tony Lee** 11:44 PM

`!cuckoo -u http://www[.]google[.]com`



CyBot

**cybot** APP 11:44 PM

Querying Cuckoo Server: `http://[REDACTED]:8090`

Function: URL Query

Success!

`task_id = 2661`

# RELEASING AT BLACK HAT USA 2019

▪ `!cuckoo -vt <hash>`



**Tony Lee** 12:18 AM

`!cuckoo -vt 57f222d8fbe0e290b4bf8eaa994ac641`



CyBot

**cybot** APP 12:18 AM

Querying Cuckoo Server: [http://\[REDACTED\]:8090](http://[REDACTED]:8090)

Function: VirusTotal File Query


Received HTTP 200. VirusTotal has the file!


Zippping the file with a password of: infected

```
{
  "task_ids": [
    2663
  ]
}
```

# RELEASING AT BLACK HAT USA 2019

▪ `!cuckoo -v <task_id>`

 **Tony Lee** 11:44 PM  
!cuckoo -v 2661

 **cybot** APP 11:44 PM  
Querying Cuckoo Server: [http://\[redacted\]:8090](http://[redacted]:8090)  
Function: View

```
{
  "task": {
    "added_on": "2019-07-11 03:44:42",
    "analysis_finished_on": null,
    "analysis_started_on": null,
    "anti_issues": null,
    "api_calls": null,
    "category": "url",
    "clock": "2019-07-11 03:44:42",
    "completed_on": null,
    "crash_issues": null,
    "custom": "",
    "domains": null,
    "dropped_files": null,
    "enforce_timeout": false,
    "errors": [],
    "files_written": null,
    "guest": {},
    "id": 2661,
```

# RELEASING AT BLACK HAT USA 2019

▪ `!cuckoo -r <task_id>`



Tony Lee 6:52 PM

!cuckoo -r 1807



cybot APP 6:52 PM

Querying Cuckoo Server: [http://\[redacted\]:8090](http://[redacted]:8090)

Function: Report Fetch

Full report most likely located here: [http://\[redacted\]:8080/analysis/1807](http://[redacted]:8080/analysis/1807)

==== INFO ====

ID: 1807

Score: 0.8 of 10.

Duration: 162 seconds

Machine Name: CyTGM\_Cuckoo\_Win7x86\_00

==== TARGET ====

url: [https://www\[.\]cylance\[.\]com](https://www[.]cylance[.]com)

category: url

==== Cuckoo Signatures ====

Allocates read-write-execute memory (usually to unpack itself) (Severity: 2)

Creates executable files on the filesystem (Severity: 2)

==== VirusTotal ====

permalink: <https://www.virustotal.com/url/7252bf65c6cbe167ed3a6ea0a7f3e882a2361c07b3451ad1e2f9956fb62f03d6/analysis/1558357226/>

scan\_date: 2019-05-20 13:00:26

positives: 0

==== Dropped Files ====

Pivot to full report

High-level info

Analyzed file or site (site is defanged)

Any Cuckoo signatures

VT hit info and link



# DEVELOPMENT

- Always on-going
- Roadmap is robust
  - Automated report writing
  - SIEM integration
  - Ticketing system integration
- Open to everyone as a community project
- Plugins written in Python
- Plugins are chat backend agnostic and will work with all platforms supported by errbot
- Code contributions go to GitHub
- New ideas welcomed!



# PRO TIPS

- If installed on Raspberry Pi

- Set a static IP

- `sudo leafpad /etc/dhcpd.conf` (at the end enter)  
`interface eth0`  
`static ip_address=<ip.ad.dr.ess>`  
`static routers=<ip.ad.dr.ess>`  
`static domain_name_servers=<ip.ad.dr.ess>`

- You may need to set US (or other regional) keyboard since this defaults to UK

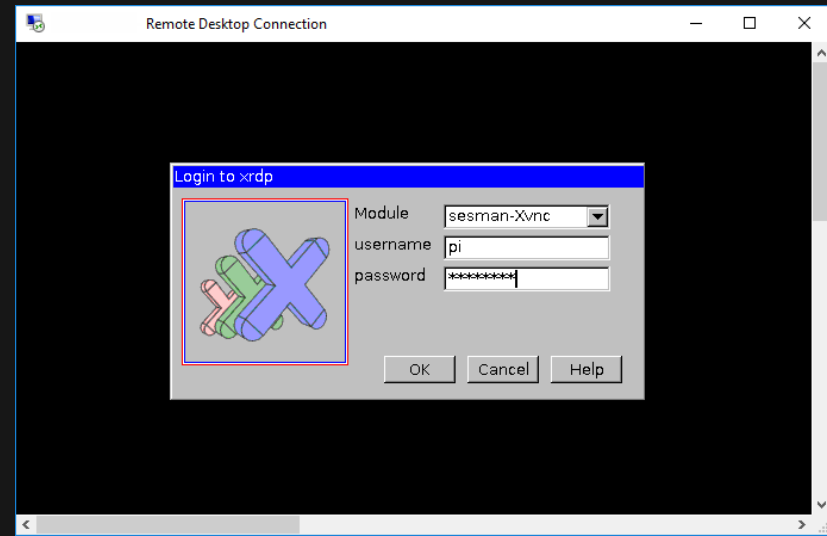
- “Start” -> Preferences -> Mouse and Keyboard Settings -> Keyboard tab -> Keyboard Layout -> United States -> English (US)

- Install remote desktop

- `sudo apt-get install vnc4server`  
▪ `sudo apt-get install xrdp`

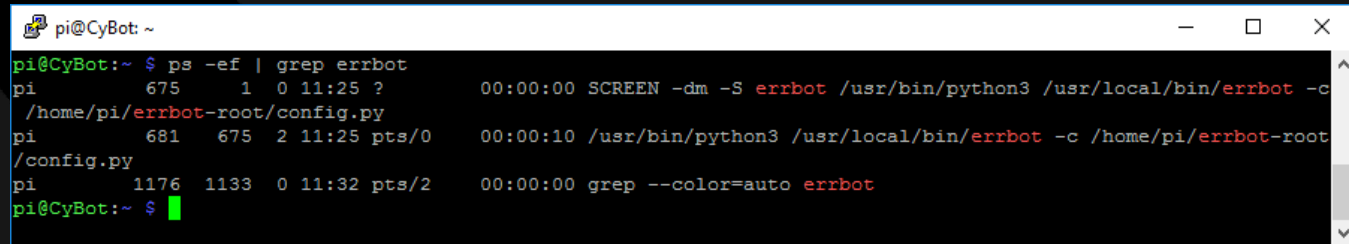
- Set SSH to autorun on boot

- `sudo update-rc.d ssh enable`



# PRO TIPS

- If installed on Raspberry Pi (cont.)
  - Set errbot to autorun on boot in a detached screen session
    - `sudo apt-get install screen`
    - `sudo leafpad /etc/rc.local`
    - Add the following line before the “exit 0” line:
      - `su - pi -c “screen -dm -S errbot /usr/bin/python3 /usr/local/bin/errbot -c /home/pi/errbot-root/config.py”`
  - Interact with the screen session by using: “screen -d -r”
  - Adjust screensaver settings easily (this will place a shortcut in preferences):
    - `sudo apt-get install xscreensaver`

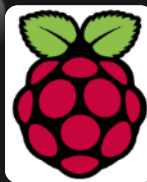
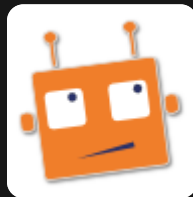


```
pi@CyBot: ~  
pi@CyBot:~ $ ps -ef | grep errbot  
pi      675      1  0 11:25 ?        00:00:00 SCREEN -dm -S errbot /usr/bin/python3 /usr/local/bin/errbot -c  
/home/pi/errbot-root/config.py  
pi      681      675  2 11:25 pts/0    00:00:10 /usr/bin/python3 /usr/local/bin/errbot -c /home/pi/errbot-root  
/config.py  
pi      1176    1133  0 11:32 pts/2    00:00:00 grep --color=auto errbot  
pi@CyBot:~ $
```

# ACKNOWLEDGEMENTS

- **Platform:**

- ErrBot - <http://errbot.io/en/latest/>
- Raspberry Pi - <https://www.raspberrypi.org/>



- **Plugins:**

- VirusTotal - <https://www.virustotal.com/>
- GeoIP – ip-api-com
- Google – safe browsing, news feeds
- Hashid – c0re
- Unshorten – unshorten.me
- Many more...

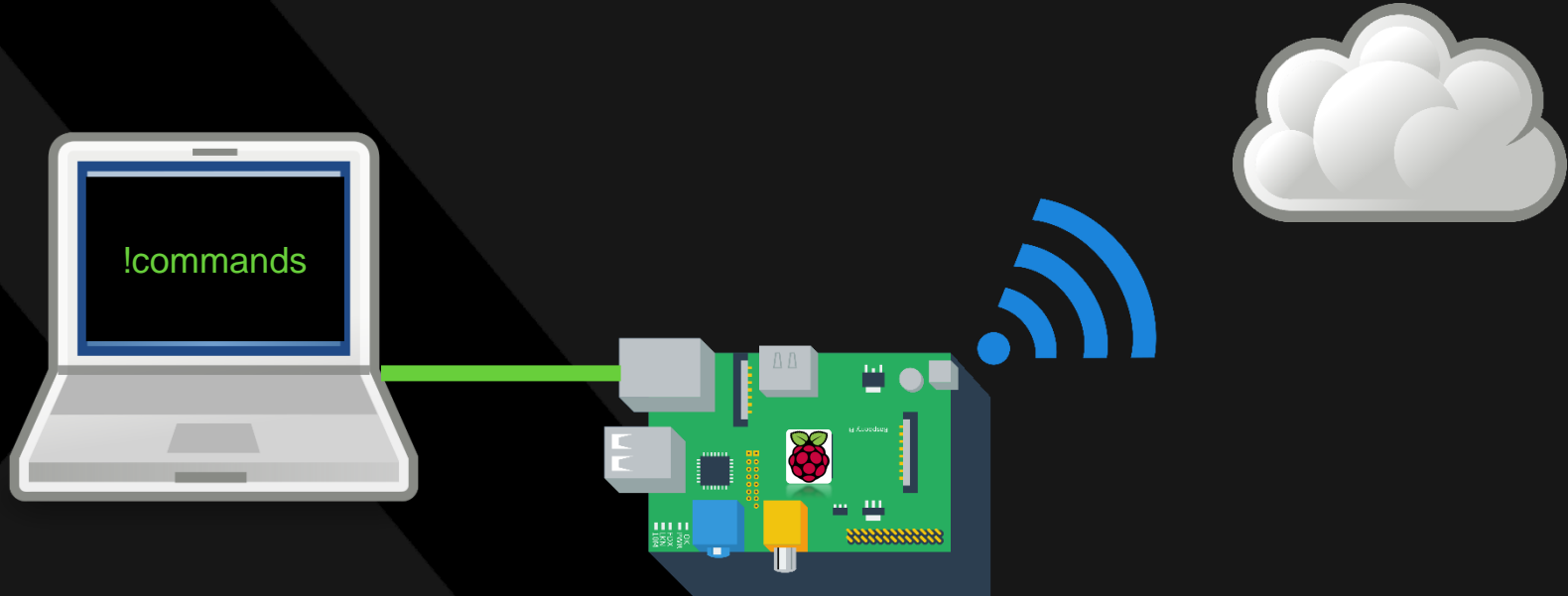


Google Safe Browsing

- **People:**

- Bill Hau, Corey White, Dennis Hanzlik, Ian Ahl, Dave Pany, Dan Dumond, Kyle Champlin

# DEMO!



The background is a blurred image of a large crowd of people, possibly at a conference or event. Overlaid on this are several dark, semi-transparent geometric shapes, including a large triangle on the left and various rectangular and trapezoidal shapes, creating a modern, abstract design.

# QUESTIONS — AND — ANSWERS



**Thank you**

# BACKUP SLIDE IN CASE LIVE DEMO TANKS

```
>>> !geoip www.google.com
"ip": "2607:f8b0:4006:81b::2004"
"country_code": "US"
"country_name": "United States"
"region_code": ""
"region_name": ""
"city": ""
"zip_code": ""
"time_zone": ""
"latitude": 37.751
"longitude": -97.822
"metro_code": 0
```

```
>>> !hashid 57f222d8fbe0e290b4bf8eaa994ac641
Analyzing '57f222d8fbe0e290b4bf8eaa994ac641'
[+] MD2
[+] MD5
[+] MD4
[+] Double MD5
[+] LM
[+] RIPEMD-128
[+] Haval-128
[+] Tiger-128
[+] Skein-256(128)
--snip--
```

```
>>> !safebrowsing ihaveaproblem.info
{'matches': [{'threatEntryType': 'URL', 'cacheDuration': '300s', 'threat': {'url': 'ihaveaproblem.info'}, 'threatType': 'SOCIAL_ENGINEERING', 'platformType': 'ANY_PLATFORM']}]}
```