Your Blacklist is Dead. Airgap Everything

The Future of Command and Control is the Cloud

$whoami

Erick Galinkin

Threat Researcher at Netskope

Better looking in selfies where I wear Lib tshirts but I don’t have a real headshot

Applied Mathematics at Johns Hopkins

Sweet title. Why are you here?

Malware authors don’t like to get caught.

Blending in with the crowd is a good way to avoid detection.

What if malware used a whitelisted domain instead of talking to an obviously sketchy server in Ukraine?

It’s notoriously difficult to detect malicious services that look like normal traffic.

SaaS - What is it good for?

Software as a Service applications are everywhere in business today and there's no end of growth in sight.

SaaS is becoming progressively more common in the enterprise. (Salesforce, Dropbox, Box, G Suite, Office 365, Slack, AWS, DocuSign, Github, Atlassian Suite, Tableau, *etc.*)

Cheaper than managing physical infrastructure, patched for you, users can access from anywhere.

Ok then - to the cloud!

Digression: Is Social Media SaaS?

From SalesForce: “Software as a service (or SaaS; pronounced /sæs/) is a way of delivering centrally hosted applications over the Internet—as a service”

Is Reddit SaaS? Is Facebook SaaS?

- You can share data over it and it’s centrally hosted so for our purposes, yes.
- It requires a login, so for our purposes, yes.

Is SaaS Good for Security?

Uhhhhh maybe.

Offloading of risk to a third party can be good. Their patching is probably better than your business

Decreased attack surface within your network

Capitalism incentivizes protecting your stuff (but doesn't always work)

Does having a bunch of whitelisted services decrease visibility?

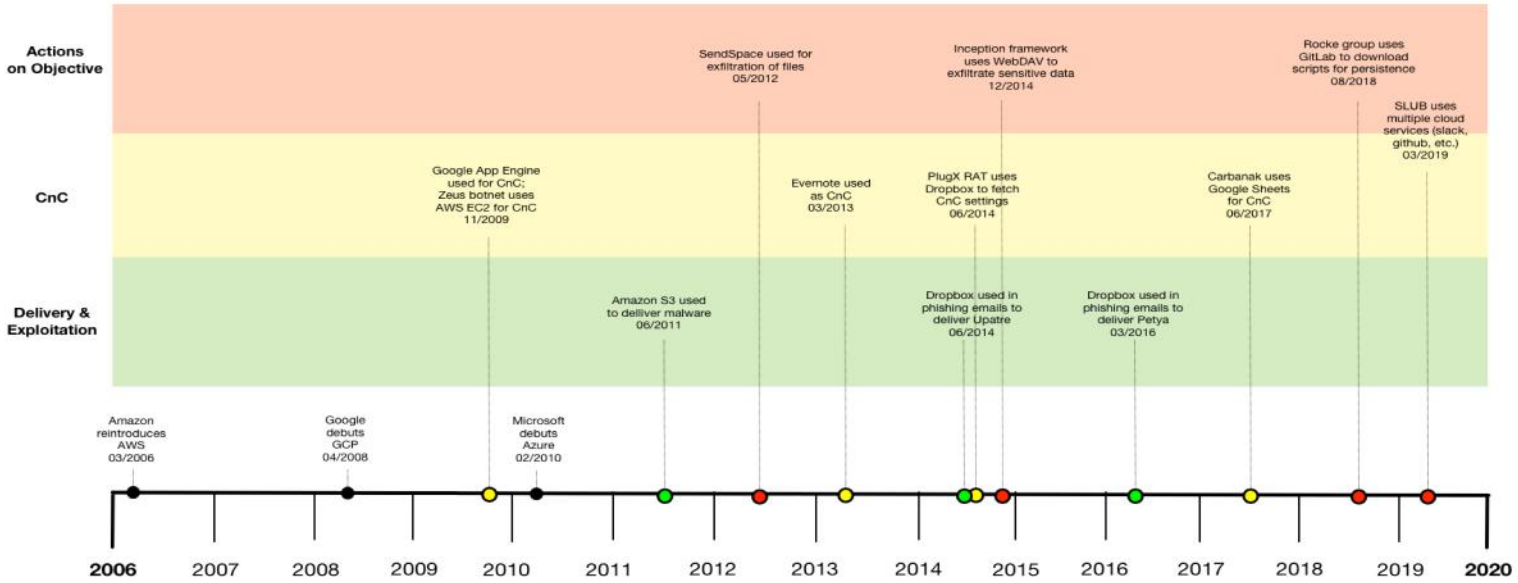
SaaS Usage by Attackers: a Blue Team Perspective

Other People Here Are Better At This

This topic could be a talk all to itself. This has been a caveat.

If you’re blue team - you absolutely need to think about SaaS services the way that you would think about exposed ports.

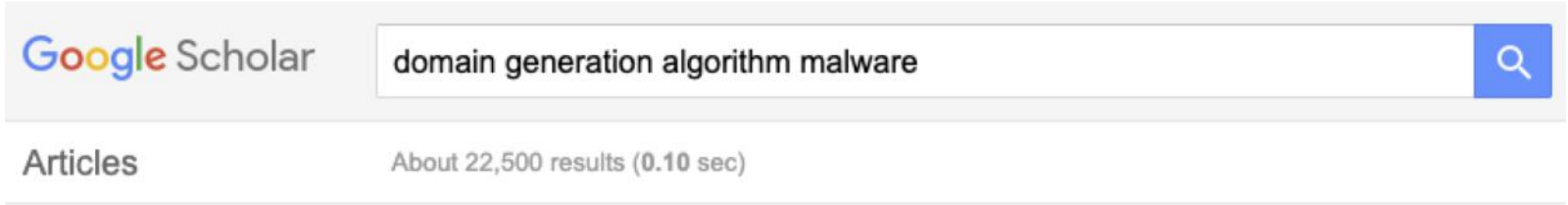
A Brief History of Malware Using the Cloud



Source for graphic: A forthcoming paper by the Netskope Threat Research Team where you might actually be able to read the text

Domain Generation Algorithms

est. 2008

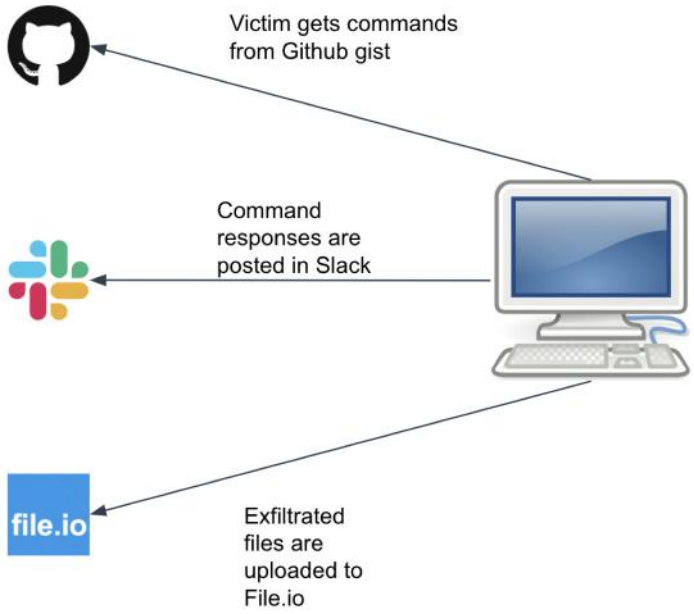


- Kraken
- Conficker
- Necurs
- GameOver (Zeus)

Case Study: SLUB

The SLUB authors removed Github from the most recent version of the backdoor - leveraging Slack far more extensively.

It continues to use file.io for file uploads



Why was it effective?

Variety of SaaS services in use (Anonymous services in particular are great!)

Built-in TLS helps evade IDS

Very sneaky - doesn't look like command and control traffic

None of these domains or IPs are likely to be blacklisted

IDS signatures depend on easily changed API keys and URI paths

Defending against SaaSy malware

- DLP solutions
- Careful monitoring
- Aggressive User Education
- Endpoint Detection

There is No New Thing Under the Sun

Treat your cloud services with the same rigor as your on-prem services.

SaaS Usage by Attackers: a Red Team Perspective

The Breaching is the Hardest Part

- Abusing lax IAM permissions
- Brute forcing logins
- Social Engineering
- Good Old Fashioned Endpoint Exploitation
- BlueKeep EternalBlue who knows what other Blues we'll see

Get by (your IDS) With a Little Help From my Friends

Downloading tools from trusted sources instead of your sketchy-ass OVH server

- Dropbox
- S3
- Pastebin
- Github

Make ‘em like Carole King

Tried and true techniques:

- Drafts with attachments in email
- Slack
- Twitter
- Dropbox

One more thing

Paying Homage

Using SaaS as command and control is not new.

- Gcat (https://github.com/byt3bl33d3r/gcat)
- Twittor (https://github.com/PaulSec/twittor)
- Slackor (https://github.com/Coalfire-Research/Slackor)
- SLUB (Real malware! Shout out to Cedric Pernet, Daniel Lunghi, Jaromir Horejsi, and Joseph C. Chen at Trend Micro. No shoutout to the malware authors.)
- But a little bit of a shoutout to the SLUB authors because I basically jacked their program flow.

Demo of the SaaSy boi

Code for SaaSy_Boi is Available!

148 lines (122 sloc) | 4.12 KB — Raw | Blame | History

```
#!/usr/bin/env python

#####################################################################
#           Agent for SaaSy_boi proof of concept code.              #
# Written by Erick Galinkin. Never use this code for anything, it #
# probably doesn't even work right.                                 #
# Agent.py can (and would have been if I were courageous enough     #
# to use GoLang) be compiled for Windows. I use a bunch of APIs     #
# that are only available on Windows, and this is only intended     #
# for use on Windows systems. As a demo. Only. Only for research. #
# Seriously - this code is for *RESEARCH* and lacks real actual     #
# malicious functionality. Ok? Ok. Thanks. Tip your bartender.      #
#####################################################################

# Imports
import os
import platform
import getpass
import apis
import utils
import sys
```

https://github.com/erickgalinkin/saas_cnc

What Have We Learned?

SaaS applications can increase your attack surface

SaaS applications move data surreptitiously

APIs make it easy to use one (or 8) services for doing bad stuff.

Props/Slops

Props:

- Slack for having a super easy-to-use REST API
- Facebook for making it very difficult to have a chat bot that abuses their TOS
- PaulSec, Coalfire, Trend Micro, and basically anyone who says “duh” to this whole presentation.
- Jenko Hwong, who made the timeline graphic for this and our paper far nicer.

Slops:

- Slack for having a super easy-to-abuse REST API
- Facebook for making it very difficult for me to have a chat bot that abuses their TOS
- My cat Dasha, who spilled my coffee and made me lose like an hour drying out my laptop.

Thanks for listening to my Talk!

Twitter: @erickgalinkin
Blog: galinked.in