# /var/log/notes

Archived: 2026-04-06 01:22:26 UTC

By Jeff White (karttoon)

Regular expressions (regex) are a language construct that allow you to define a search pattern. The flexibility of this language allows you to craft search patterns for tons of practical applications, including passive identification of network traffic. Specifically, they can allow you to pattern match on URL's so that you may quickly identify malicious sites frequently used by malware command and control (C2), domain generation algorithms (DGA's), and other such activities.

I fell in love with using regex as a defensive tool while doing incident response many years ago. The depth of control they provide naturally lends itself to the forensic, analyst, and responder lines of work. This blog may be old hat to most blue teamers out there, but if not, hopefully it serves as an educational resource on how you can use data to build PCRE's for network defense.

Over the course of this blog, I'll cover developing Perl compatible regex (PCRE) for the Emotet banking malware download URL's and develop PCRE's that encompass multiple campaigns that can then be used on a proxy device (blocking), in a SIEM (identification), or whatever system you have that supports utilizing these expressions. Emotet is a great candidate for review as it has varying domain structures that are ripe for pattern matching. I'll walk you through how I develop these PCRE's, along with refining them, and then finally how they can be vetted for false-positives (FP) to make them ready for production.

Throughout the blog, I'll be using a Python script I wrote called pcre_check to assit with the analysis. Essentially, all the tool does is take a parameter for a file containing your PCRE's, a parameter for a file containing the URL's, and then some flags for how to display the pattern matches and misses. This is helpful for the rapid development of PCRE's because, more often than naught, you find yourself in the midst of developing these when the shit has hit the fan...or at least I always did.

I'll be focusing solely on URL's in this example; however, on the off chance you're not familiar with regex, keep in mind that a myriad of tools, all the way down at the byte level and up to the application level that I'll be covering here can utilize regex. You should absolutely learn the basics at least as it's something that can be a life saver in your daily toolbox.

Before I get too much further in, here are a couple of helpful links, that I find myself constantly visiting, which you may find useful if you want to review or build your own PCRE's. I'll try to explain the regex syntax and logic as I go but I'll assume you know the basic structure of the language. If not, hit the references below.

- https://regex101.com/ - Lets you test a PCRE (or some other flavors of regex) against a set of strings you provide on the fly with color and syntax highlighting. It provides extremely helpful explanations that tell you how your PCRE is being evaluated so you can adjust as needed.

- http://www.regular-expressions.info/ - This site probably has everything you ever wanted to know about the regex language. A super handy quick reference for when you forget some of the nuances and syntax.

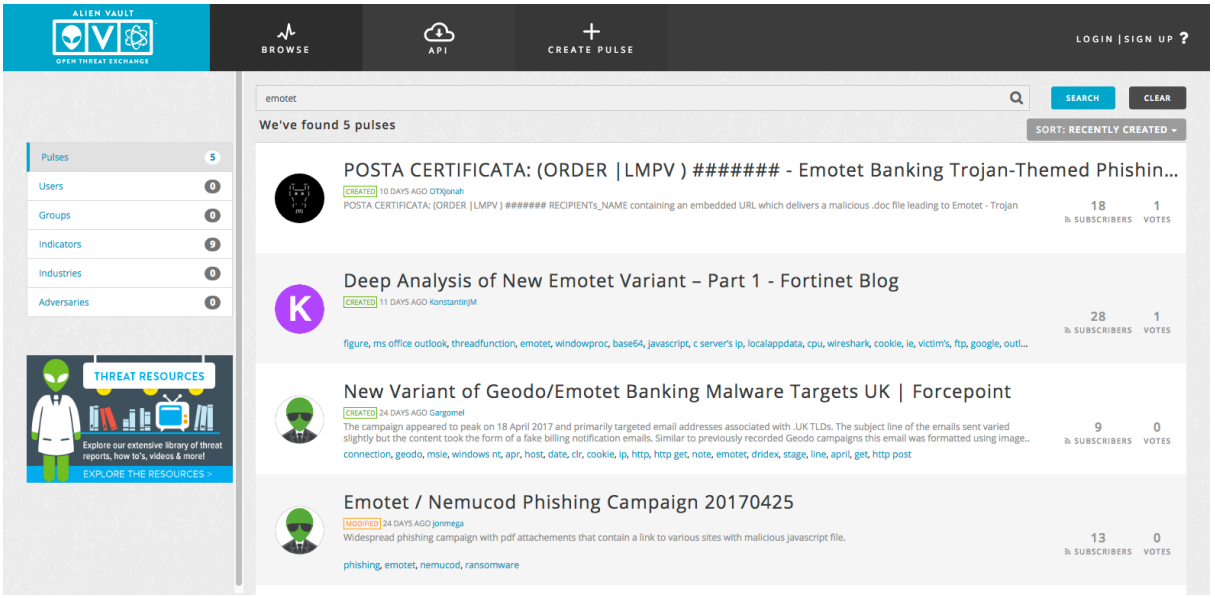
This will be a long blog, and a little free flowing, as I develop these while enumerating step-by-step. Below are some jumps so you can skip around as needed.


- Creating your sample corpus
- Identifying patterns and enumeration
  - Round 1
  - Round 2
  - Round 3
  - Round 4
  - Round 5
  - Round 6
- Refining the rules
- Vetting the rules
- Final product


#### Initial Sample Corpus

The Emotet banking malware download locations have a lot of different URL structures across their different campaigns. It's been popping up on my radar more and more lately so I want to try and enumerate the patterns here to further expand what I can catch. That being said, the very first thing I need to do is collect a decent samples of the various campaigns so that I can begin to try and match them. Prior to my current $dayjob, I'd approach this by hitting up multiple blogs from researchers or security companies and compile the URL set. When I didn't have access to systems that made this task fairly trivial, I would frequently build them from the below resources.

- Alient Vault Open Threat Exchange (OTX) - An awesome aggregation project that lets you pivot around various reports, blogs, and events based on keywords and extract what you need. Below is a screenshot showing a search for "emotet"; each of those contain IOC's for URL's you can copy out to build your list.




- Malware don't need Coffee - Kafeine's site is more focused on exploit kits but almost always had a handful of URL's of interest and sometimes links to raw URL dumps on Github.
- Malware Traffic Analysis - MalwareTraffic's site is heavily focused on exploit kits and e-mail based threats, but almost always includes domains/URL's as well.

Usually just Googling the threat name, "Emotet domains", bring you to sites like this one which have links to Pastebin posts containing loads of samples. The more the better but in general, in my experience, I'd say between 15-30 URL's is usually enough to make a solid base for an individual pattern and then you can tweak it during the false-positive (FP) checking phase.

I've placed 696 Emotet URL's on Github which you can use to follow along or throw in a blocklist.

#### Pattern Recognition / Enumeration

Once you have a decent sample set, the next step is to analyze the data and look for patterns. I'll show the various changes to the PCRE's as I analyze the URL's and you can see how they evolve into the final product after each iteration. To better illustrate this, I'll just focus on the last 20 URL's at a time but normally I'll have open 3 terminals: top window editing the URL file, middle window with pcre_check output, bottom window editing the PCRE file. This layout allows me to quickly modify and validate changes on the fly and significantly reducing the time to turnaround.

Below is the first run of the script showing that none of the URLs matched and truncated to the last 20.

#### Round 1

$ python pcre_check.py -p emotet_pcres -u emotet_urls -n [+] NO HITS [+] http://12back.com/dw3wz-ue164-qqv/ http://4glory.net/p7lrq-s191-iv/ ... http://www.melodywriters.com/INVOICE-864339-98261/ http://www.prodzakaz.com.ua/H27560xzwsS/ http://www.stellaimpianti.it/download2467/ http://www.stepstonedev.com/field/download7812/ http://www.surreycountycleaners.com/t5wx-x064-mzdb/ http://www.voloskof.net/Sn83160EngQs/ http://www.wildweek.com/EDHFR-08-77623-document-May-04-2017/

http://www.ziyufang.studio/project/wp-content/plugins/nprojects/download5337/ http://wyskocil.de/ORDER-525808-73297/ http://xionglutions.com/NDKBS-51-84402-document-May-03-2017/ http://xionglutions.com/wl7dh-uf201-asnw/ http://xyphoid.com/RRT-13279129.dokument/ http://xyphoid.com/SCANNED/MM3431UCNPCEZRO/ http://yildiriminsaat.com.tr/JCV-71815736.dokument/ http://zahahadidmiami.com/K38258Q/ http://zeroneed.com/FNN-40446899.dokument/ http://ziarahsutera.com/5377959590/ http://zonasacra.com/zH83293YizhQ/ http://zvarga.com/15-12-07/CUST-9405847-8348/ http://zypern-aktiv.de/wp-content/plugins/wordfence/img9re-a789-stz/

There are a couple of things that jump out immediately on the first review.

- Domains seem unrelated to the URL path, they are most likely compromised sites.
- The final path can be multiple levels down, so I'll need to account for this.
- At least 6 different variations can be seen out of the gate.

For each of the PCRE, I've grown accustomed to starting them with the below structure.

^http:\/\/[^\x2F]+\/

This matches any line that begins ("^") with "http://" followed by any characters, except ("[^ ]") forward slash ("\x2F"), up to the first foward slash. This ensures we match the domain regardless of what TLD or subdomains may be present.

For ease of illustration, I'm going to group the variations and break them down individually.

**[Group 01]**

http://www.surreycountycleaners.com/t5wx-x064-mzdb/ http://xionglutions.com/wl7dh-uf201-asnw/ http://zypern-aktiv.de/wp-content/plugins/wordfence/img9re-a789-stz/

For this pattern, we have 4-5 alpha(lower)numeric, dash, 4-5 alpha(lower)numeric, dash, 3-4 alpha(lower). We'll also want to account for the last line which has the path multiple levels in. We can accomplish this by putting our "[^\x2F]+\/" section in a group and saying the group can repeat one or more times (eg match everything between the forward slashes until the last one, where our pattern is).

^http:\/\/([^\x2F]+\/)+[a-z0-9]{4,5}-[a-z0-9]{4,5}-[a-z]{3,4}\/$

**[Group 02]**

http://www.melodywriters.com/INVOICE-864339-98261/ http://wyskocil.de/ORDER-525808-73297/ http://zvarga.com/15-12-07/CUST-9405847-8348/

This next group appears to use a word in caps, dash, 6-7 numbers, dash, 4-5 numbers. We'll need to account for the subpaths again as well. In this case, I prefer to group full words instead of using a character range, which helps for trying to be false-positive adverse.

^http:\/\/([^\x2F]+\/)+(INVOICE|ORDER|CUST)-[0-9]{6,7}-[0-9]{4,5}\/$

**[Group 03]**

http://www.prodzakaz.com.ua/H27560xzwsS/ http://www.voloskof.net/Sn83160EngQs/ http://xyphoid.com/SCANNED/MM3431UCNPCEZRO/ http://zahahadidmiami.com/K38258Q/ http://ziarahsutera.com/5377959590/ http://zonasacra.com/zH83293YizhQ/

I feel this group may end up getting split later. We have one URL which is purely numerical and then two which have no lowercase letters. We'll cross that bridge as we look at more samples, if necessary. Another thing to note is that this group has a very weak pattern in that it is very generic, which means it will likely match a lot of legitimate URL's and not hold up during FP testing. We'll cross that bridge when we get to it as well.

For now, it's a mix of 7-15 alphanumeric characters.

^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{7,15}\/$

**[Group 04]**

http://xyphoid.com/RRT-13279129.dokument/ http://yildiriminsaat.com.tr/JCV-71815736.dokument/ http://zeroneed.com/FNN-40446899.dokument/

This one, and the next three, all look pretty straight forward: 3 alpha(upper), dash, 8 numbers, period, "dokument" string.

^http:\/\/([^\x2F]+\/)+[A-Z]{3}-[0-9]{8}\.dokument\/$

**[Group 05]**

http://www.wildweek.com/EDHFR-08-77623-document-May-04-2017/ http://xionglutions.com/NDKBS-51-84402-document-May-03-2017/

Similarly, very structured (which is good for us): 5 alpha(upper), dash, 2 numbers, dash, 5 numbers, dash, "document" string, dash, "May" string, dash, 2 numbers, dash, "2017" string. I've defaulted to using "2017" as a string since it aligns with their usage of it as a date so it seems unlikely to change.

^http:\/\/([^\x2F]+\/)+[A-Z]{5}-[0-9]{2}-[0-9]{5}-document-May-[0-9]{2}-2017\/$

**[Group 06]**

http://www.stellaimpianti.it/download2467/ http://www.stepstonedev.com/field/download7812/ http://www.ziyufang.studio/project/wp-content/plugins/nprojects/download5337/

The string "download", 4 numbers.

^http:\/\/([^\x2F]+\/)+download[0-9]{4}\/$

I'll throw these into the emotet_pcres file and see how each performs against our target data set of known-bad Emotet sites.

[+] FOUND [+] Count: 24/696 Comment: Group 01 - [ t5wx-x064-mzdb ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-z0-9]{4,5}-[a-z0-9]{4,5}-[a-z]{3,4}\/$ [+] FOUND [+] Count: 43/696 Comment: Group 02 - [ INVOICE-864339-98261 ] PCRE: ^http:\/\/([^\x2F]+\/)+(INVOICE|ORDER|CUST)-[0-9]{6,7}-[0-9]{4,5}\/$ [+] FOUND [+] Count: 177/696 Comment: Group 03 - [ H27560xzwsS ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{7,15}\/$ [+] FOUND [+] Count: 30/696 Comment: Group 04 - [ RRT-13279129.dokument ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Z]{3}-[0-9]{8}\.dokument\/$ [+] FOUND [+] Count: 24/696 Comment: Group 05 - [ EDHFR-08-77623-document-May-04-2017 ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Z]{5}-[0-9]{2}-[0-9]{5}-document-May-[0-9]{2}-2017\/$ [+] FOUND [+] Count: 62/696 Comment: Group 06 - [ download2467 ] PCRE: ^http:\/\/([^\x2F]+\/)+download[0-9]{4}\/$

Pretty low across the board except for group 3, which is the one I mentioned is too loose to begin with. From here on out, if I don't list a particular group, it implies there was no change to the PCRE.

#### Round 2

The next 20 URL's are below.

http://web2present.com/Invoice-538878-14610/ http://webbmfg.com/krupy/gallery2/g2data/LUqc663BAyN333-HoO/ http://webbsmail.co.uk/DIDE-19-85247-document-May-04-2017/ http://webergy.co.uk/15-14-47/Cust-0910279-3981/ http://webics.org/Cust-951068-69554/ http://websajt.nu/ap6ohc-au152-urttp/ http://wescographics.com/17-40-07/Invoice-5558936-1201/ http://whiteroofradio.com/YD796MJO974-NNW/ http://wightman.cc/ipa0oab-j490-keap/ http://wilstu.com/hHiDSaaP03Y95TIGpIUS4Aa/ http://wingitproductions.org/NUDA-X-52454-DE/ http://wlrents.com/CUST.-Document-YDI-04-GQ389557/ http://wnyil.org/wnyil_transfer/Ups__com__WebTracking__tracknum__4DFH74180493688150/ORDER.-Document-SY-92-E736730/ http://wolffy.net/17-00-07/Invoice-9545415-1483/ http://wortis.com/CH760Wcv003-Luh/ http://www.anti-corruption.su/Cust-3708876-8210/ http://www.anti-corruption.su/TNO-59-97413-document-May-04-2017/ http://www.babyo.com.mx/Invoice-583156-73417/ http://www.doodle.tj/yW1NZ-sh00-cH/ http://zypern-aktiv.de/wp-content/plugins/wordfence/img9re-a789-stz/

It looks like we have a few new groups as well. I'll attempt to highlight in red the changes to the PCRE's which might make the changes clearer.

**[Group 01] - [ t5wx-x064-mzdb ]**

http://websajt.nu/ap6ohc-au152-urttp/ http://wightman.cc/ipa0oab-j490-keap/ http://www.doodle.tj/yW1NZ-sh00-cH/ http://zypern-aktiv.de/wp-content/plugins/wordfence/img9re-a789-stz/

You'll note that the third one now introduces capital letters; it's possible this is a separate campaign but I'll circle back to this later during review. The main changes will be the addition of the capital letters and adjustment on the ranges, which will likely be the case for the rest of the groups.

OLD: ^http:\/\/([^\x2F]+\/)+[a-z0-9]{4,5}-[a-z0-9]{4,5}-[a-z]{3,4}\/$ NEW: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{4,7}-[a-z0-9]{4,5}-[a-z]{2,5}\/$

**[Group 02] - [ INVOICE-864339-98261 ]**

http://web2present.com/Invoice-538878-14610/ http://webergy.co.uk/15-14-47/Cust-0910279-3981/ http://webics.org/Cust-951068-69554/ http://wescographics.com/17-40-07/Invoice-5558936-1201/ http://wolffy.net/17-00-07/Invoice-9545415-1483/ http://www.anti-corruption.su/Cust-3708876-8210/ http://www.babyo.com.mx/Invoice-583156-73417/

New strings "Invoice" and "Cust".

OLD: ^http:\/\/([^\x2F]+\/)+(INVOICE|ORDER|CUST)-[0-9]{6,7}-[0-9]{4,5}\/$ NEW: ^http:\/\/([^\x2F]+\/)+(INVOICE|ORDER|CUST|Invoice|Cust)-[0-9]{6,7}-[0-9]{4,5}\/$

**[Group 03] - [ H27560xzwsS ]**

http://wilstu.com/hHiDSaaP03Y95TIGpIUS4Aa/

Range adjustment (making this one even more useless).

OLD: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{7,15}\/$ NEW: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{7,23}\/$

**[Group 05] - [ EDHFR-08-77623-document-May-04-2017 ]**

http://webbsmail.co.uk/DIDE-19-85247-document-May-04-2017/ http://www.anti-corruption.su/TNO-59-97413-document-May-04-2017/

Range adjustment.

OLD: ^http:\/\/([^\x2F]+\/)+[A-Z]{5}-[0-9]{2}-[0-9]{5}-document-May-[0-9]{2}-2017\/$ NEW: ^http:\/\/([^\x2F]+\/)+[A-Z]{3,5}-[0-9]{2}-[0-9]{5}-document-May-[0-9]{2}-2017\/$

**[Group 07] - [ LUqc663BAyN333-HoO ]**

http://webbmfg.com/krupy/gallery2/g2data/LUqc663BAyN333-HoO/ http://whiteroofradio.com/YD796MJO974-NNW/ http://wortis.com/CH760Wcv003-Luh/

This cluser is defined by one dash towards the end: 11-14 alphanumeric, dash, 3 alpha.

^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{11,14}-[a-zA-Z]{3}\/$

**[Group 08] - [ NUDA-X-52454-DE ]**

http://wingitproductions.org/NUDA-X-52454-DE/

Only one sample so I'll match it exactly, 4 alpha(upper), dash, 1 alpha(upper), dash, 5 numbers, dash, 2 alpha(upper).

^http:\/\/([^\x2F]+\/)+[A-Z]{4}-[A-Z]{1}-[0-9]{5}-[A-Z]{2}\/$

**[Group 09] - [ CUST.-Document-YDI-04-GQ389557 ]**

http://wlrents.com/CUST.-Document-YDI-04-GQ389557/
http://wnyil.org/wnyil_transfer/Ups__com__WebTracking__tracknum__4DFH74180493688150/ORDER.-Document-SY-92-E736730/

Similar to Group 2: same word choice, period, dash, "Document" string, dash, 2-3 alpha(upper), dash, 2 numbers, dash, 7-8 alpha(upper)numeric.

^http:\/\/([^\x2F]+\/)+(CUST|ORDER)\.-Document-[A-Z]{2,3}-[0-9]{2}-[A-Z0-9]{7,8}\/$

Note that the delta in the output after each group is just something I've included after the fact to show the progress for the blog.

[+] FOUND [+] Count: 66/696 (+42) Comment: [Group 01] - [ t5wx-x064-mzdb ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{4,7}-[a-z0-9]{4,5}-[a-z]{2,5}\/$ [+] FOUND [+] Count: 80/696 (+37) Comment: [Group 02] - [ INVOICE-864339-98261 ] PCRE: ^http:\/\/([^\x2F]+\/)+(INVOICE|ORDER|CUST|Invoice|Cust)-[0-9]{6,7}-[0-9]{4,5}\/$ [+] FOUND [+] Count: 190/696 (+13) Comment: [Group 03] - [ H27560xzwsS ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{7,23}\/$ [+] FOUND [+] Count: 30/696 Comment: [Group 04] - [ RRT-13279129.dokument ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Z]{3}-[0-9]{8}\.dokument\/$ [+] FOUND [+] Count: 59/696 (+35) Comment: [Group 05] - [ EDHFR-08-77623-document-May-04-2017 ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Z]{3,5}-[0-9]{2}-[0-9]{5}-document-May-[0-9]{2}-2017\/$ [+] FOUND [+] Count: 62/696 Comment: [Group 06] - [ download2467 ] PCRE: ^http:\/\/([^\x2F]+\/)+download[0-9]{4}\/$ [+] FOUND [+] Count: 15/696 Comment: [Group 07] - [ LUqc663BAyN333-HoO ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{11,14}-[a-zA-Z]{3}\/$ [+] FOUND [+] Count: 3/696 Comment: [Group 08] - [ NUDA-X-52454-DE ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Z]{4}-[A-Z]{1}-[0-9]{5}-[A-Z]{2}\/$ [+] FOUND [+] Count: 20/696 Comment: [Group 09] - [ CUST.-Document-YDI-04-GQ389557 ] PCRE: ^http:\/\/([^\x2F]+\/)+(CUST|ORDER)\.-Document-[A-Z]{2,3}-[0-9]{2}-[A-Z0-9]{7,8}\/$

#### Round 3

The next set of 20 URL's.

http://theocforrent.com/BG-47535325/zp3x-r88-wuh.view/ http://thepogs.net/rs4eG-Md93-FSZV/ http://thesubservice.com/ORDER.-Document-9543529814/ http://theuntoldsorrow.co.uk/ORDER.-XI-80-UY913942/ http://tiger12.com/TGA-48-76252-doc-May-04-2017/ http://timmadden.com.au/qzw1s-wc740-m/ http://toppprogramming.com/Cust-8328499631/ http://tpsystem.net/TaVS391hyCaD623-dJ/ http://transfinity.co.uk/sam/fathers-day/htdocs/b2m-qp699-jxmln/ http://tridentii.com/OY-30676027.dokument/ http://tscoaching.co.uk/l1R-q60-pe/ http://uncover.jp/XwXL806QaDN792-jr/ http://uncover.jp/r-2psl-vo440-lz.doc/ http://visionsoflightphotography.com/FRMLW-RNT-41482-DE/ http://visuals.com/CUST.-VT-38-RH422386/ http://voxellab.com/BBM-07-75350-doc-May-04-2017/ http://vspacecreative.co.uk/O2-view-report-818/c1o-jn07-er.view/ http://wayanad.net/xhW017TRfP646-z/ http://wb0rur.com/ZGAG-59-63863-doc-May-05-2017/ http://www.doodle.tj/yW1NZ-sh00-cH/

One new variant in this set.

**[Group 01] - [ t5wx-x064-mzdb ]**

http://thepogs.net/rs4eG-Md93-FSZV/ http://timmadden.com.au/qzw1s-wc740-m/ http://transfinity.co.uk/sam/fathers-day/htdocs/b2m-qp699-jxmln/ http://tscoaching.co.uk/l1R-q60-pe/ http://www.doodle.tj/yW1NZ-sh00-cH/

Range adjustment and additiona case changes.

OLD: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{4,7}-[a-z0-9]{4,5}-[a-z]{2,5}\/$ NEW: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{4,7}-[a-zA-Z0-9]{4,5}-[a-zA-Z]{1,5}\/$

**[Group 02] - [ INVOICE-864339-98261 ]**

http://toppprogramming.com/Cust-8328499631/

This could be a different campaign as it breaks from the double-dashes but it's so similar to group 2 that I'll leave it for now and possibly revisit.

The second dash I'll make optional which should allow the lowest ranges of the numerical sections to match. I'll use an optinal capturing group ("(-)?") for the second dash. Effectively creating a capture group and then using the "?" value after will cause the group to match between zero and one time, thus becoming optional.

OLD: ^http:\/\/([^\x2F]+\/)+(INVOICE|ORDER|CUST|Invoice|Cust)-[0-9]{6,7}-[0-9]{4,5}\/$ NEW: ^http:\/\/([^\x2F]+\/)+(INVOICE|ORDER|CUST|Invoice|Cust)-[0-9]{6,7}(-)?[0-9]{4,5}\/$

**[Group 04] - [ RRT-13279129.dokument ]**

http://tridentii.com/OY-30676027.dokument/

Range adjustment.

OLD: ^http:\/\/([^\x2F]+\/)+[A-Z]{3}-[0-9]{8}\.dokument\/$ NEW: ^http:\/\/([^\x2F]+\/)+[A-Z]{2,3}-[0-9]{8}\.dokument\/$

**[Group 05] - [ EDHFR-08-77623-document-May-04-2017 ]**

http://tiger12.com/TGA-48-76252-doc-May-04-2017/ http://voxellab.com/BBM-07-75350-doc-May-04-2017/ http://wb0rur.com/ZGAG-59-63863-doc-May-05-2017/

Add "doc" string to grouping.

OLD: ^http:\/\/([^\x2F]+\/)+[A-Z]{3,5}-[0-9]{2}-[0-9]{5}-document-May-[0-9]{2}-2017\/$ NEW: ^http:\/\/([^\x2F]+\/)+[A-Z]{3,5}-[0-9]{2}-[0-9]{5}-(document|doc)-May-[0-9]{2}-2017\/$

**[Group 07] - [ LUqc663BAyN333-HoO ]**

http://tpsystem.net/TaVS391hyCaD623-dJ/ http://uncover.jp/XwXL806QaDN792-jr/ http://wayanad.net/xhW017TRfP646-z/

Range adjustment.

OLD: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{11,14}-[a-zA-Z]{3}\/$ NEW: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{11,15}-[a-zA-Z]{1,3}\/$

**[Group 08] - [ NUDA-X-52454-DE ]**

http://visionsoflightphotography.com/FRMLW-RNT-41482-DE/

Range adjustment.

OLD: ^http:\/\/([^\x2F]+\/)+[A-Z]{4}-[A-Z]{1}-[0-9]{5}-[A-Z]{2}\/$ NEW: ^http:\/\/([^\x2F]+\/)+[A-Z]{4,5}-[A-Z]{1,3}-[0-9]{5}-[A-Z]{2}\/$

**[Group 09] - [ CUST.-Document-YDI-04-GQ389557 ]**

http://thesubservice.com/ORDER.-Document-9543529814/ http://theuntoldsorrow.co.uk/ORDER.-XI-80-UY913942/ http://visuals.com/CUST.-VT-38-RH422386/

Couple of things going on here.

New grouping of words for second part and first entry is only numerical without dashes, which looks similar to the new entry for Group 2. To account for these, I'll use optional capturing groups again to build around them. It makes the rule slightly less accurate but with the other anchors in it, I think it'll still be fairly unique enough to not FP.

NEW: ^http:\/\/([^\x2F]+\/)+(CUST|ORDER)\.-Document-[A-Z]{2,3}-[0-9]{2}-[A-Z0-9]{7,8}\/$ OLD: ^http:\/\/([^\x2F]+\/)+(CUST|ORDER)\.-(Document|XI|VT)((-[A-Z]{2,3})?-[0-9]{2})?-[A-Z0-9]{7,10}\/$

**[Group 10] - [ zp3x-r88-wuh.view ]**

http://theocforrent.com/BG-47535325/zp3x-r88-wuh.view/ http://uncover.jp/r-2psl-vo440-lz.doc/ http://vspacecreative.co.uk/O2-view-report-818/c1o-jn07-er.view/

The "doc" and "view" ones may be different campaigns but, again, I'll lump them together for now and will separate at the end if necessary: 1-4 alpha(lower)numeric, dash, 3-4 alpha(lower)numeric, dash, optional 5 alpha(lower)numeric, dash, 2-3 alpha(lower), period, group "view" or "doc" strings.

^http:\/\/([^\x2F]+\/)+[a-z0-9]{1,4}-[a-z0-9]{3,4}(-[a-z0-9]{5})?-[a-z]{2,3}\.(view|doc)\/$

The pcre_check output shows decent coverage improvements.

[+] FOUND [+] Count: 93/696 (+27) Comment: [Group 01] - [ t5wx-x064-mzdb ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{4,7}-[a-zA-Z0-9]{4,5}-[a-zA-Z]{1,5}\/$ [+] FOUND [+] Count: 89/696 (+9) Comment: [Group 02] - [ INVOICE-864339-98261 ] PCRE: ^http:\/\/([^\x2F]+\/)+(INVOICE|ORDER|CUST|Invoice|Cust)-[0-9]{6,7}(-)?[0-9]{4,5}\/$ [+] FOUND [+] Count: 190/696 Comment: [Group 03] - [ H27560xzwsS ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{7,23}\/$ [+] FOUND [+] Count: 56/696 (+26) Comment: [Group 04] - [ RRT-

13279129.dokument ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Z]{2,3}-[0-9]{8}\.dokument\/$ [+] FOUND [+] Count: 79/696 (+20) Comment: [Group 05] - [ EDHFR-08-77623-document-May-04-2017 ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Z]{3,5}-[0-9]{2}-[0-9]{5}-(document|doc)-May-[0-9]{2}-2017\/$ [+] FOUND [+] Count: 62/696 Comment: [Group 06] - [ download2467 ] PCRE: ^http:\/\/([^\x2F]+\/)+download[0-9]{4}\/$ [+] FOUND [+] Count: 43/696 (+28) Comment: [Group 07] - [ LUqc663BAyN333-HoO ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{11,15}-[a-zA-Z]{1,3}\/$ [+] FOUND [+] Count: 10/696 (+7) Comment: [Group 08] - [ NUDA-X-52454-DE ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Z]{4,5}-[A-Z]{1,3}-[0-9]{5}-[A-Z]{2}\/$ [+] FOUND [+] Count: 31/696 (+11) Comment: [Group 09] - [ CUST.-Document-YDI-04-GQ389557 ] PCRE: ^http:\/\/([^\x2F]+\/)+(CUST|ORDER)\.-(Document|XI|VT)((-[A-Z]{2,3})?-[0-9]{2})?-[A-Z0-9]{7,10}\/$ [+] FOUND [+] Count: 3/696 Comment: [Group 10] - [ zp3x-r88-wuh.view ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-z0-9]{1,4}-[a-z0-9]{3,4}(-[a-z0-9]{5})?-[a-z]{2,3}\.(view|doc)\/$

**Round 4**

The next 20 sites.

http://pinoypiper.com/Sz1Mr-H23-Xw/ http://proiecte-pac.ro/ORDER.-5883789520/ http://proprints.dk/Rech-74779857260/ http://pulmad.ee/B6y-Fb95-NMW/ http://redkitecottages.com/Cust-Document-VMH-46-TJ804065/ http://reichertgroup.com/d0r-tl410-cxa/ http://sgbusiness.co.uk/YM-57911235-document-May-03-2017/ http://sign1.no/dhl___status___2668292851/ http://sloan3d.com/Cust-Document-WMV-26-EW054554/ http://stacibockman.com/g2c-o179-pocja/ http://streamingair.com/i0A-St59-m/ http://sublevel3.us/G7n-Gh58-y/ http://superalumnos.net/php/ORDER.-HW-84-Y947883/ http://technetemarketing.com/CUST.-8520279770/ http://teed.ru/YG-47124992/bc7za-l30-v.view/ http://texasbrits.com/m3s-r623-x/ http://thegilbertlawoffice.com/m-9q-d054-gu.doc/ http://thenursesagent.com/ORDER.-9592209302/ http://transfinity.co.uk/sam/fathers-day/htdocs/b2m-qp699-jxmln/ http://tscoaching.co.uk/l1R-q60-pe/

One new variant sticks out, otherwise business as usual.

**[Group 01] - [ t5wx-x064-mzdb ]**

http://pinoypiper.com/Sz1Mr-H23-Xw/ http://pulmad.ee/B6y-Fb95-NMW/ http://reichertgroup.com/d0r-tl410-cxa/ http://stacibockman.com/g2c-o179-pocja/ http://streamingair.com/i0A-St59-m/ http://sublevel3.us/G7n-Gh58-y/ http://texasbrits.com/m3s-r623-x/ http://transfinity.co.uk/sam/fathers-day/htdocs/b2m-qp699-jxmln/ http://tscoaching.co.uk/l1R-q60-pe/

Half of the 20 are for this group. Just some small range adjustments.

OLD: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{4,7}-[a-zA-Z0-9]{4,5}-[a-zA-Z]{1,5}\/$ NEW: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{3,7}-[a-zA-Z0-9]{3,5}-[a-zA-Z]{1,5}\/$

**[Group 02] - [ INVOICE-864339-98261 ]**

http://proiecte-pac.ro/ORDER.-5883789520/ http://proprints.dk/Rech-74779857260/ http://technetemarketing.com/CUST.-8520279770/ http://thenursesagent.com/ORDER.-9592209302/

It should be apparent now that Group 2 and 9 have a bit of overlap and I was going to wait till the end to course correct; however, I feel it's just too much at this point so I'm going to split it so the ones above, and previously matched in both groups, with the "ORDER" and "CUST" strings followed by 10 digits are a new unique group. That means I need to edit Group 2 and 9 to avoid these and the simplest way of doing that is removing the previous optional dash, making it absolutely required. See Group 9 and 12 for further iteration details.

OLD: ^http:\/\/([^\x2F]+\/)+(INVOICE|ORDER|CUST|Invoice|Cust)-[0-9]{6,7}(-)?[0-9]{4,5}\/$ NEW: ^http:\/\/([^\x2F]+\/)+(INVOICE|ORDER|CUST|Invoice|Cust)-[0-9]{6,7}-[0-9]{4,5}\/$

**[Group 05] - [ EDHFR-08-77623-document-May-04-2017 ]**

http://sgbusiness.co.uk/YM-57911235-document-May-03-2017/

This new one breaks from the two parts separated by a dash. I can add the dash to the character list and up the range, or I can opt for a optional grouping and up the range. I'm going to do the latter for the reason that it keeps the structure in tact; for this, I'm not as worried about FP's due to the ending part of the pattern being fairly unique.

OLD: ^http:\/\/([^\x2F]+\/)+[A-Z]{3,5}-[0-9]{2}-[0-9]{5}-(document|doc)-May-[0-9]{2}-2017\/$ NEW: ^http:\/\/([^\x2F]+\/)+[A-Z]{2,5}(-[0-9]{2})?-[0-9]{5,10}-(document|doc)-May-[0-9]{2}-2017\/$

**[Group 09] - [ CUST.-Document-YDI-04-GQ389557 ]**

http://redkitecottages.com/Cust-Document-VMH-46-TJ804065/ http://sloan3d.com/Cust-Document-WMV-26-EW054554/ http://superalumnos.net/php/ORDER.-HW-84-Y947883/

Similar to Group 2, I'm going to reverse course on the optional groupings so that the 10 digits are not captured. To account for the new variants in Group 9, I'm adding an optional grouping for the period after the first word and for the "Document" string, then moving the others back into the A-Z grouping that followed.

OLD: ^http:\/\/([^\x2F]+\/)+(CUST|ORDER)\.-(Document|XI|VT)((-[A-Z]{2,3})?-[0-9]{2})?-[A-Z0-9]{7,10}\/$ NEW: ^http:\/\/([^\x2F]+\/)+(CUST|ORDER|Cust)(.)?(-Document)?-[A-Z]{2,3}-[0-9]{2}-[A-Z0-9]{7,10}\/$

**[Group 10] - [ zp3x-r88-wuh.view ]**

http://thegilbertlawoffice.com/m-9q-d054-gu.doc/

Range adjustment.

OLD: ^http:\/\/([^\x2F]+\/)+[a-z0-9]{1,4}-[a-z0-9]{3,4}(-[a-z0-9]{5})?-[a-z]{2,3}\.(view|doc)\/$ NEW: ^http:\/\/([^\x2F]+\/)+[a-z0-9]{1,4}-[a-z0-9]{3,4}(-[a-z0-9]{4,5})?-[a-z]{2,3}\.(view|doc)\/$

**[Group 11] - [ dhl___status___2668292851 ]**

http://sign1.no/dhl___status___2668292851/

Not much to work with yet so it's fairly static.

^http:\/\/([^\x2F]+\/)+dhl___status___[0-9]{10}\/$

**[Group 12] - [ ORDER.-5883789520 ]**

Previous set: http://thesubservice.com/ORDER.-Document-9543529814/ http://toppprogramming.com/Cust-8328499631/ Current set: http://proiecte-pac.ro/ORDER.-5883789520/ http://proprints.dk/Rech-74779857260/ http://technetemarketing.com/CUST.-8520279770/ http://thenursesagent.com/ORDER.-9592209302/

Looking at the data in Group 2 and 9, this pattern will have: string grouping of "ORDER", "RECH", "CUST", "Cust", optional period, dash, optional "Document" string, 10-11 numbers. By the way, "rech" is shorthand for "rechnung", which is German for "bill" - you see these variations quite a bit in phishing campaigns as they focus on different regions.

^http:\/\/([^\x2F]+\/)+(ORDER|Rech|CUST|Cust)(.)?(-Document)?-[0-9]{10,11}\/$

Next iteration below.

[+] FOUND [+] Count: 127/696 (+34) Comment: [Group 01] - [ t5wx-x064-mzdb ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{3,7}-[a-zA-Z0-9]{3,5}-[a-zA-Z]{1,5}\/$ [+] FOUND [+] Count: 80/696 (-9) Comment: [Group 02] - [ INVOICE-864339-98261 ] PCRE: ^http:\/\/([^\x2F]+\/)+(INVOICE|ORDER|CUST|Invoice|Cust)-[0-9]{6,7}-[0-9]{4,5}\/$ [+] FOUND [+] Count: 190/696 Comment: [Group 03] - [ H27560xzwsS ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{7,23}\/$ [+] FOUND [+] Count: 56/696 Comment: [Group 04] - [ RRT-13279129.dokument ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Z]{2,3}-[0-9]{8}\.dokument\/$ [+] FOUND [+] Count: 86/696 (+7) Comment: [Group 05] - [ EDHFR-08-77623-document-May-04-2017 ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Z]{2,5}(-[0-9]{2})?-[0-9]{5,10}-(document|doc)-May-[0-9]{2}-2017\/$ [+] FOUND [+] Count: 62/696 Comment: [Group 06] - [ download2467 ] PCRE: ^http:\/\/([^\x2F]+\/)+download[0-9]{4}\/$ [+] FOUND [+] Count: 43/696 Comment: [Group 07] - [ LUqc663BAyN333-HoO ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{11,15}-[a-zA-Z]{1,3}\/$ [+] FOUND [+] Count: 10/696 Comment: [Group 08] - [ NUDA-X-52454-DE ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Z]{4,5}-[A-Z]{1,3}-[0-9]{5}-[A-Z]{2}\/$ [+] FOUND [+] Count: 36/696 (+5) Comment: [Group 09] - [ CUST.-Document-YDI-04-GQ389557 ] PCRE: ^http:\/\/([^\x2F]+\/)+(CUST|ORDER|Cust)(.)?(-Document)?-[A-Z]{2,3}-[0-9]{2}-[A-Z0-9]{7,10}\/$ [+] FOUND [+] Count: 3/696 Comment: [Group 10] - [ zp3x-r88-wuh.view ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-z0-9]{1,4}-[a-z0-9]{3,4}(-[a-z0-9]{4,5})?-[a-z]{2,3}\.(view|doc)\/$ [+] FOUND [+] Count: 3/696 Comment: [Group 11] - [ dhl___status___2668292851 ] PCRE: ^http:\/\/([^\x2F]+\/)+dhl___status___[0-9]{10}\/$ [+] FOUND [+] Count: 31/696 Comment: [Group 12] - [ ORDER.-5883789520 ] PCRE: ^http:\/\/([^\x2F]+\/)+(ORDER|Rech|CUST|Cust)(.)?(-Document)?-[0-9]{10,11}\/$

**Round 5**

Since there are only 31 URL's left I'm just going to add them all here and close out this phase.

http://akhmerov.com/AuHffUo4L1BcEmca0BW5e4UtI/ http://albrightfinancial.com/gescanntes-Dokument-66764196575/ http://anjep.com/TBWEV-YCAP-91327-DE/ http://arroyave.net/Rech-K-682-GO1130/ http://beowulf7.com/kgcee/ http://bitach.com/RIJW-FNFE-86299-DE/ http://bobrow.com/ito-6r-w193-pkr.doc/ http://boningue.com/g843enx500-Jh/ http://carriedavenport.com/Scan-58146582290/

http://davidberman.com/gescanntes-Dokument-85218870046/ http://dentaltravelpoland.co.uk/NUN-63376893/b4fe-nn88-s.view/ http://donnjo.com/Rechnung-IOOY-776-LUV2894/ http://frossweddingcollections.co.uk/qdu-7p-wi523-hgnt.doc/ http://froufrouandthomas.co.uk/c644kNg297-uy/ http://gabrielramos.com.br/lxu-3h-ip079-zgmg.doc/ http://genxvisual.com/U494KHq064-VK/ http://gestion-arte.com.ar/CLCJY-EMIE-76216-DE/ http://imnet.ro/gcxbh/ http://johncarta.com/jexaag/ http://kowalenko.ca/D603ImA780-xxJ/ http://kratiroff.com/Scan-62799108494/ http://lapetitenina.com/eyym/ http://magmaprod.com.br/FcmUZ9GGTFaq2SYC5HTuFgc4v7/ http://masmp.com/rby-4c-rp108-sqq.doc/ http://missgypsywhitemoon.com.au/ismpce/ http://music111.com/VAQT-DYBC-27274-DE/ http://myhorses.ca/lb8TApg9aZI6PP5RWRAIdmfU/ http://onlineme.w04.wh-2.com/LD-36666076/ir5r-mu75-h.view/ http://phoneworx.co.uk/HLqwOU1uNQ7rWLWkXW6VoMheZf/ http://teed.ru/YG-47124992/bc7za-l30-v.view/ http://thegilbertlawoffice.com/m-9q-d054-gu.doc/

**[Group 03] - [ H27560xzwsS ]**

http://akhmerov.com/AuHffUo4L1BcEmca0BW5e4UtI/ http://beowulf7.com/kgcee/ http://imnet.ro/gcxbh/ http://johncarta.com/jexaag/ http://lapetitenina.com/eyym/ http://magmaprod.com.br/FcmUZ9GGTFaq2SYC5HTuFgc4v7/ http://myhorses.ca/lb8TApg9aZI6PP5RWRAIdmfU/ http://phoneworx.co.uk/HLqwOU1uNQ7rWLWkXW6VoMheZf/

I'll adjust the ranges on this one but you can see from the above that it looks like two distinct campaigns. I have no doubt now that there will be more in this grouping but since it's almost over 200 URL's I'll review the entire set at the end.

OLD: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{7,23}\/$ NEW: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{4,26}\/$

**[Group 07] - [ LUqc663BAyN333-HoO ]**

http://boningue.com/g843enx500-Jh/ http://froufrouandthomas.co.uk/c644kNg297-uy/ http://genxvisual.com/U494KHq064-VK/ http://kowalenko.ca/D603ImA780-xxJ/

Range adjustment.

OLD: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{11,15}-[a-zA-Z]{1,3}\/$ NEW: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{10,15}-[a-zA-Z]{1,3}\/$

**[Group 08] - [ NUDA-X-52454-DE ]**

http://anjep.com/TBWEV-YCAP-91327-DE/ http://bitach.com/RIJW-FNFE-86299-DE/ http://gestion-arte.com.ar/CLCJY-EMIE-76216-DE/ http://music111.com/VAQT-DYBC-27274-DE/

Range adjustment. Curious these all end with "DE" too, possibly region based given the "Rech" stuff seen previously; will follow-up after.

OLD: ^http:\/\/([^\x2F]+\/)+[A-Z]{4,5}-[A-Z]{1,3}-[0-9]{5}-[A-Z]{2}\/$ NEW: ^http:\/\/([^\x2F]+\/)+[A-Z]{4,5}-[A-Z]{1,4}-[0-9]{5}-[A-Z]{2}\/$

**[Group 09] - [ CUST.-Document-YDI-04-GQ389557 ]**

http://arroyave.net/Rech-K-682-GO1130/ http://donnjo.com/Rechnung-IOOY-776-LUV2894/

Added "Rech" and "Rechnung" to initial string grouping along with expanding some ranges.

OLD: ^http:\/\/([^\x2F]+\/)+(CUST|ORDER|Cust)(.)?(-Document)?-[A-Z]{2,3}-[0-9]{2}-[A-Z0-9]{7,10}\/$ NEW: ^http:\/\/([^\x2F]+\/)+(CUST|ORDER|Cust|Rech|Rechnung)(.)?(-Document)?-[A-Z]{1,4}-[0-9]{2,3}-[A-Z0-9]{6,10}\/$

**[Group 10] - [ zp3x-r88-wuh.view ]**

http://bobrow.com/ito-6r-w193-pkr.doc/ http://dentaltravelpoland.co.uk/NUN-63376893/b4fe-nn88-s.view/ http://frossweddingcollections.co.uk/qdu-7p-wi523-hgnt.doc/ http://gabrielramos.com.br/lxu-3h-ip079-zgmg.doc/ http://masmp.com/rby-4c-rp108-sqq.doc/ http://onlineme.w04.wh-2.com/LD-36666076/ir5r-mu75-h.view/ http://teed.ru/YG-47124992/bc7za-l30-v.view/ http://thegilbertlawoffice.com/m-9q-d054-gu.doc/

Range adjustment.

OLD: ^http:\/\/([^\x2F]+\/)+[a-z0-9]{1,4}-[a-z0-9]{3,4}(-[a-z0-9]{4,5})?-[a-z]{2,3}\.(view|doc)\/$ NEW: ^http:\/\/([^\x2F]+\/)+[a-z0-9]{1,5}-[a-z0-9]{2,4}(-[a-z0-9]{4,5})?-[a-z]{1,4}\.(view|doc)\/$

**[Group 12] - [ ORDER.-5883789520 ]**

http://albrightfinancial.com/gescanntes-Dokument-66764196575/ http://carriedavenport.com/Scan-58146582290/ http://davidberman.com/gescanntes-Dokument-85218870046/ http://kratiroff.com/Scan-62799108494/

Added "gescanntes" to initial string grouping (this is Dutch for "Scanned") and "Scan". Added "Dokument" to second optional grouping.

OLD: ^http:\/\/([^\x2F]+\/)+(ORDER|Rech|CUST|Cust)(.)?(-Document)?-[0-9]{10,11}\/$ NEW: ^http:\/\/([^\x2F]+\/)+(ORDER|Rech|CUST|Cust|gescanntes|Scan)(.)?(-Document|-Dokument)?-[0-9]{10,11}\/$

Alright, now I've cleared all of the remaining matches.

[+] FOUND [+] Count: 127/696 Comment: [Group 01] - [ t5wx-x064-mzdb ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{3,7}-[a-zA-Z0-9]{3,5}-[a-zA-Z]{1,5}\/$ [+] FOUND [+] Count: 80/696 Comment: [Group 02] - [ INVOICE-864339-98261 ] PCRE: ^http:\/\/([^\x2F]+\/)+(INVOICE|ORDER|CUST|Invoice|Cust)-[0-9]{6,7}-[0-9]{4,5}\/$ [+] FOUND [+] Count: 199/696 (+9) Comment: [Group 03] - [ H27560xzwsS ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{4,26}\/$ [+] FOUND [+] Count: 56/696 Comment: [Group 04] - [ RRT-13279129.dokument ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Z]{2,3}-[0-9]{8}\.dokument\/$ [+] FOUND [+] Count: 86/696 Comment: [Group 05] - [ EDHFR-08-77623-document-May-04-2017 ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Z]{2,5}(-[0-9]{2})?-[0-9]{5,10}-(document|doc)-May-[0-9]{2}-2017\/$ [+] FOUND [+] Count: 62/696 Comment: [Group 06] - [ download2467 ] PCRE: ^http:\/\/([^\x2F]+\/)+download[0-9]{4}\/$ [+] FOUND [+] Count: 47/696 (+4) Comment: [Group 07] - [ LUqc663BAyN333-HoO ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{10,15}-[a-zA-Z]{1,3}\/$ [+] FOUND [+] Count: 14/696 (+4) Comment: [Group 08] - [ NUDA-X-52454-DE ] PCRE:

^http:\/\/([^\x2F]+\/)+[A-Z]{4,5}-[A-Z]{1,4}-[0-9]{5}-[A-Z]{2}\/$ [+] FOUND [+] Count: 38/696 (+2) Comment: [Group 09] - [ CUST.-Document-YDI-04-GQ389557 ] PCRE: ^http:\/\/([^\x2F]+\/)+(CUST|ORDER|Cust|Rech|Rechnung)(.)?(-Document)?-[A-Z]{1,4}-[0-9]{2,3}-[A-Z0-9]{6,10}\/$ [+] FOUND [+] Count: 11/696 (+1) Comment: [Group 10] - [ zp3x-r88-wuh.view ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-z0-9]{1,5}-[a-z0-9]{2,4}(-[a-z0-9]{4,5})?-[a-z]{1,4}\.(view|doc)\/$ [+] FOUND [+] Count: 3/696 Comment: [Group 11] - [ dhl___status___2668292851 ] PCRE: ^http:\/\/([^\x2F]+\/)+dhl___status___[0-9]{10}\/$ [+] FOUND [+] Count: 35/696 (+4) Comment: [Group 12] - [ ORDER.-5883789520 ] PCRE: ^http:\/\/([^\x2F]+\/)+(ORDER|Rech|CUST|Cust|gescanntes|Scan)(.)?(-Document|-Dokument)?-[0-9]{10,11}\/$

**Round 6**

The next step is to validate the matches with the "-s" flag in pcre_check. This will show all of the respective matches under each PCRE. For this phase, I just eyeball it to make sure there is no overlap and what's expected in each group is present.

All of the PCRE's look solid except Group 3, which I already mentioned would need more TLC, as it overlaps with other PCRE's.

For Group 3, I'm going to visually break these down. I'll put 5 examples under each sub-grouping to show how I separated them. Some are very good for matching while others will just have to be left behind. TAKE NOTE BAD GUYS, BEING GENRIC IS GOOD, UNIQUE SNOWFLAKES ARE THE FIRST AGAINST THE WALL.

**[Group 03] - [ dhl/paket/com/pkp/appmanager/8376315127 ]**

http://8kindsoffun.com/dhl/paket/com/pkp/appmanager/8376315127/
http://balletopia.org/dhl/paket/com/pkp/appmanager/7293445574/
http://cnwconsultancy.com/dhl/paket/com/pkp/appmanager/0622636111/
http://cookieco.com/dhl/paket/com/pkp/appmanager/8333287922/
http://cspdx.com/dhl/paket/com/pkp/appmanager/6213914600/

I think thins one would have stood out earlier had it not been clobbered by the previous PCRE. The path is very unique and ends with 10 digits. This PCRE will replace the old one for Group 3 and the other new ones will start at Group 13.

^http:\/\/([^\x2F]+\/)+dhl\/paket\/com\/pkp\/appmanager\/[0-9]{10}\/$

**[Group 13] - [ 6572646300 ]**

http://alfareklama.cz/6572646300/ http://algicom.net/6673413599/ http://bourdin.name/0014489972/ http://carbitech.net/dhl/2354409458/ http://dsltech.co.uk/0217183208/ ... http://oscartvazquez.com/DHL24/15382203695/

I'm going to create a PCRE for this one but I don't expect it to live past the FP check. There is one that stands off from the rest here with 11 numbers instead of 10 - it may be that I just don't have enough samples to account for

that campaign. Finally, I'll need to exclude the previous set of matches which also end with 10 digits. To do this, I'll use a negative lookbehind to ensure once we match 10 digits, "appmanager" was not in the URL path.

^http:\/\/([^\x2F]+\/)+(?<!appmanager\/)[0-9]{10,11}\/$

**[ alpha(lower) ]**

http://aifesdespets.fr/kkrxtsmodw/ http://beowulf7.com/kgcee/ http://bunngalow.com/injeutznnb/ http://carbofilms.com/cms/wp-content/upgrade/jcnfkvken/ http://dolphinrunvb.com/yozypdznpb/

I don't see any good patterns in this set or the next one.

**[ alpha(lower)numeric ]**

http://benard.ca/z49641l/ http://jaqua.us/hid4kiwcvd84fljkpqpl/ http://krakhud.pl/rguen0ebxndrci41frworbr/ http://micromatrices.com/qwh7zxijifxsnxg20mlwa/ http://patu.ch/bgrvm2wqpjw74hz/

**[Group 14] [ SCANNED/RZ7498WEXEZB ]**

http://icaredentalstudio.com/APE88743TZ/ http://lbcd.se/MFV09235UA/ http://lucasliftruck.com/SCANNED/RZ7498WEXEZB/ http://meanconsulting.com/K44975X/ http://sentios.lt/W95941C/ http://triadesolucoes.com.br/SCANNED/RBA6517MHPKCZDEX/ http://xyphoid.com/SCANNED/MM3431UCNPCEZRO/ http://zahahadidmiami.com/K38258Q/

This group was characterized by alpha(upper)numeric, which normally wouldn't be worth pattern matching, but I can see two patterns in the above that may be worth entertaining. For Group 14, I'll match on the URL's with "SCANNED" string in the path and the unique placement of the digits within the string: 2-3 alpha(upper), 4 digits, 6-9 alpha(upper).

^http:\/\/([^\x2F]+\/)+SCANNED\/[A-Z]{2,3}[0-9]{4}[A-Z]{6,9}\/$

**[Group 15] [ K44975X ]**

http://meanconsulting.com/K44975X/ http://sentios.lt/W95941C/ http://zahahadidmiami.com/K38258Q/

For Group 15, I'll match on 1 alpha(upper), 5 digits, 1 alpha(upper). The non-matched ones in the previous Group 14 may be an expanded part of this campaign but it's such a weak PCRE and prone to FP that I'm not going to bother with it. It's highly likely to not make the final cut either way.

^http:\/\/([^\x2F]+\/)+[A-Z]{1}[0-9]{5}[A-Z]{1}\/$

**[ alphanumeric long 18-26 ]**

http://akhmerov.com/AuHffUo4L1BcEmca0BW5e4UtI/ http://arosa.nl/crm/xs2ckmwotgcml95cxdhbo/ http://crosslink.ca/nWlKL3PdKyi1goahyZfbNr/ http://ideaswebstudio.com/v3mzbzaink00sndmyz/ http://infojass.com/gvtsl7ddrnjkupn50pp/

Nothing jumps out at me that would make for a good PCRE. It has a similar structure of alpha, digit, alpha but the ranges are very broad which makes it highly prone to FP again.

**[ alphanumeric short 7-14 ]**

http://akirmak.com/QhS33472le/ http://austinaaron.com/eCjH94174LaN/ http://campanus.cz/N6571iwA/ http://carolsgardeninn.com/vX94098JvVJ/ http://cdoprojectgraduation.com/eaSz15612O/ ... http://www.alfredomartinez.com.mx/Afz3999lDtz/ http://www.kreodesign.pl/test/O77405ccSC/ http://www.prodzakaz.com.ua/H27560xzwsS/ http://www.voloskof.net/Sn83160EngQs/ http://zonasacra.com/zH83293YizhQ/

This next one follows the same pattern I identified for Group 15: 1-5 alphanumeric, 4-5 digits, 1-5 alphanumeric. I'll just update Group 15 and see how it fairs in the FP check, but for what it's worth, it does match every single entry in this category which had 30+.

OLD: ^http:\/\/([^\x2F]+\/)+[A-Z]{1}[0-9]{5}[A-Z]{1}\/$ NEW: ^http:\/\/([^\x2F]+\/)+[A-Za-z]{1,4}[0-9]{4,5}[a-zA-Z]{1,5}\/$

#### Refinement

Now that everything is clustered together, I'll do one final visual inspection to see if any other patterns jump out that allow us to tighten the rules up and avoid FP's.

**[Group 01] - [ t5wx-x064-mzdb ]**

http://12back.com/dw3wz-ue164-qqv/ http://4glory.net/p7lrq-s191-iv/ http://aconai.fr/v4OZ-PR72-gtS/ http://adamkranitz.com/gqj5ijg-y250-ex/ http://allisonhibbard.com/x4b-th601-m/

In Group 1, we can actually refine this a bit once you see the underlying pattern. Almost every part of this one changed so I'll just go back over it: 1-3 alpha, 1 digit, 1-3 alpha, dash, 1-2 alpha, 2-3 digit, dash, 1-5 alpha.

OLD: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{3,7}-[a-zA-Z0-9]{3,5}-[a-zA-Z]{1,5}\/$ NEW: ^http:\/\/([^\x2F]+\/)+[a-zA-Z]{1,3}[0-9]{1}[a-zA-Z]{1,3}-[a-zA-Z]{1,2}[0-9]{2,3}-[a-zA-Z]{1,5}\/$

**[Group 07] - [ LUqc663BAyN333-HoO ]**

http://agenity.com/EAVx829uahI723-tv/ http://argoinf.com/YFSR334KgXCe907-z/ http://artmedieval.net/RK415njzzR555-p/ http://autoradio.com.br/fRq804tvz270-tWa/ http://belief-systems.com/obn247eaC420-Z/

In Group 7, the first part of the pattern can be refined: 1-4 alphanumeric, 3 digits, 1-5 alphanumeric, 3 digits.

OLD: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{10,15}-[a-zA-Z]{1,3}\/$ NEW: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{1,4}[0-9]{3}[a-zA-Z]{1,5}[0-9]{3}-[a-zA-Z]{1,3}\/$

**[Group 08] - [ NUDA-X-52454-DE ]**

http://altius.co.in/EJZB-T-66361-DE/ http://anjep.com/TBWEV-YCAP-91327-DE/ http://aquarthe.com/AIUO-P-70826-DE/ http://bitach.com/RIJW-FNFE-86299-DE/ http://cliftonsecurities.co.uk/YJTX-NMO-51102-DE/

In Group 8 they all end with "DE" so I'll convert that part to a static string.

OLD: ^http:\/\/([^\x2F]+\/)+[A-Z]{4,5}-[A-Z]{1,4}-[0-9]{5}-[A-Z]{2}\/$ NEW: ^http:\/\/([^\x2F]+\/)+[A-Z]{4,5}-[A-Z]{1,4}-[0-9]{5}-DE\/$

**[Group 09] - [ CUST.-Document-YDI-04-GQ389557 ]**

http://archabits.com/ORDER.-AXN-60-X400251/ http://arrosio.com.ar/ORDER.-Document-SF-41-F318806/ http://arroyave.net/Rech-K-682-GO1130/ http://avenueevents.co.uk/Cust-PBP-03-D683320/ http://babyo.com.mx/Cust-Document-KEQ-04-FF065857/

In Group 9, every entry entry ends with 1-3 alpha(upper) followed by 4-6 digits.

OLD: ^http:\/\/([^\x2F]+\/)+(CUST|ORDER|Cust|Rech|Rechnung)(.)?(-Document)?-[A-Z]{1,4}-[0-9]{2,3}-[A-Z0-9]{6,10}\/$ NEW: ^http:\/\/([^\x2F]+\/)+(CUST|ORDER|Cust|Rech|Rechnung)(.)?(-Document)?-[A-Z]{1,4}-[0-9]{2,3}-[A-Z]{1,3}[0-9]{4,6}\/$

The final run for the PCRE's before FP testing.

[+] FOUND [+] Count: 127/696 Comment: [Group 01] - [ t5wx-x064-mzdb ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-zA-Z]{1,3}[0-9]{1}[a-zA-Z]{1,3}-[a-zA-Z]{1,2}[0-9]{2,3}-[a-zA-Z]{1,5}\/$ [+] FOUND [+] Count: 80/696 Comment: [Group 02] - [ INVOICE-864339-98261 ] PCRE: ^http:\/\/([^\x2F]+\/)+(INVOICE|ORDER|CUST|Invoice|Cust)-[0-9]{6,7}-[0-9]{4,5}\/$ [+] FOUND [+] Count: 29/696 (changed to new pattern) Comment: [Group 03] - [ dhl/paket/com/pkp/appmanager/8376315127 ] PCRE: ^http:\/\/([^\x2F]+\/)+dhl\/paket\/com\/pkp\/appmanager\/[0-9]{10}\/$ [+] FOUND [+] Count: 56/696 Comment: [Group 04] - [ RRT-13279129.dokument ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Z]{2,3}-[0-9]{8}\.dokument\/$ [+] FOUND [+] Count: 86/696 Comment: [Group 05] - [ EDHFR-08-77623-document-May-04-2017 ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Z]{2,5}(-[0-9]{2})?-[0-9]{5,10}-(document|doc)-May-[0-9]{2}-2017\/$ [+] FOUND [+] Count: 62/696 Comment: [Group 06] - [ download2467 ] PCRE: ^http:\/\/([^\x2F]+\/)+download[0-9]{4}\/$ [+] FOUND [+] Count: 47/696 Comment: [Group 07] - [ LUqc663BAyN333-HoO ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{1,4}[0-9]{3}[a-zA-Z]{1,5}[0-9]{3}-[a-zA-Z]{1,3}\/$ [+] FOUND [+] Count: 14/696 Comment: [Group 08] - [ NUDA-X-52454-DE ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Z]{4,5}-[A-Z]{1,4}-[0-9]{5}-DE\/$ [+] FOUND [+] Count: 38/696 Comment: [Group 09] - [ CUST.-Document-YDI-04-GQ389557 ] PCRE: ^http:\/\/([^\x2F]+\/)+(CUST|ORDER|Cust|Rech|Rechnung)(.)?(-Document)?-[A-Z]{1,4}-[0-9]{2,3}-[A-Z]{1,3}[0-9]{4,6}\/$ [+] FOUND [+] Count: 11/696 Comment: [Group 10] - [ zp3x-r88-wuh.view ] PCRE: ^http:\/\/([^\x2F]+\/)+[a-z0-9]{1,5}-[a-z0-9]{2,4}(-[a-z0-9]{4,5})?-[a-z]{1,4}\.(view|doc)\/$ [+] FOUND [+] Count: 3/696 Comment: [Group 11] - [ dhl___status___2668292851 ] PCRE: ^http:\/\/([^\x2F]+\/)+dhl___status___[0-9]{10}\/$ [+] FOUND [+] Count: 35/696 Comment: [Group 12] - [ ORDER.-5883789520 ] PCRE: ^http:\/\/([^\x2F]+\/)+(ORDER|Rech|CUST|Cust|gescanntes|Scan)(.)?(-Document|-Dokument)?-[0-9]{10,11}\/$ [+] FOUND [+] Count: 15/696 Comment: [Group 13] - [ 6572646300 ] PCRE: ^http:\/\/([^\x2F]+\/)+(?<!appmanager\/)[0-9]{10,11}\/$ [+] FOUND [+] Count: 3/696 Comment: [Group 14] [ SCANNED/RZ7498WEXEZB ] PCRE: ^http:\/\/([^\x2F]+\/)+SCANNED\/[A-Z]{2,3}[0-9]{4}[A-Z]

{6,9}\/$ [+] FOUND [+] Count: 60/696 Comment: [Group 15] [ K44975X ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Za-z]{1,4}[0-9]{4,5}[a-zA-Z]{1,5}\/$

That leaves only 30 URL's that I was unable to reliably match - not too shabby! You can find the output of the pcre_check script showing the matches and non-matches HERE.

The current PCRE list is below.

^http:\/\/([^\x2F]+\/)+[a-zA-Z]{1,3}[0-9]{1}[a-zA-Z]{1,3}-[a-zA-Z]{1,2}[0-9]{2,3}-[a-zA-Z]{1,5}\/$ [Group 01] - [ t5wx-x064-mzdb ] ^http:\/\/([^\x2F]+\/)+(INVOICE|ORDER|CUST|Invoice|Cust)-[0-9]{6,7}-[0-9]{4,5}\/$ [Group 02] - [ INVOICE-864339-98261 ] ^http:\/\/([^\x2F]+\/)+dhl\/paket\/com\/pkp\/appmanager\/[0-9]{10}\/$ [Group 03] - [ dhl/paket/com/pkp/appmanager/8376315127 ] ^http:\/\/([^\x2F]+\/)+[A-Z]{2,3}-[0-9]{8}\.dokument\/$ [Group 04] - [ RRT-13279129.dokument ] ^http:\/\/([^\x2F]+\/)+[A-Z]{2,5}(-[0-9]{2})?-[0-9]{5,10}-(document|doc)-May-[0-9]{2}-2017\/$ [Group 05] - [ EDHFR-08-77623-document-May-04-2017 ] ^http:\/\/([^\x2F]+\/)+download[0-9]{4}\/$ [Group 06] - [ download2467 ] ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{1,4}[0-9]{3}[a-zA-Z]{1,5}[0-9]{3}-[a-zA-Z]{1,3}\/$ [Group 07] - [ LUqc663BAyN333-HoO ] ^http:\/\/([^\x2F]+\/)+[A-Z]{4,5}-[A-Z]{1,4}-[0-9]{5}-DE\/$ [Group 08] - [ NUDA-X-52454-DE ] ^http:\/\/([^\x2F]+\/)+(CUST|ORDER|Cust|Rech|Rechnung)(.)?(-Document)?-[A-Z]{1,4}-[0-9]{2,3}-[A-Z]{1,3}[0-9]{4,6}\/$ [Group 09] - [ CUST.-Document-YDI-04-GQ389557 ] ^http:\/\/([^\x2F]+\/)+[a-z0-9]{1,5}-[a-z0-9]{2,4}(-[a-z0-9]{4,5})?-[a-z]{1,4}\.(view|doc)\/$ [Group 10] - [ zp3x-r88-wuh.view ] ^http:\/\/([^\x2F]+\/)+dhl___status___[0-9]{10}\/$ [Group 11] - [ dhl___status___2668292851 ] ^http:\/\/([^\x2F]+\/)+(ORDER|Rech|CUST|Cust|gescanntes|Scan)(.)?(-Document|-Dokument)?-[0-9]{10,11}\/$ [Group 12] - [ ORDER.-5883789520 ] ^http:\/\/([^\x2F]+\/)+(?<!appmanager\/)[0-9]{10,11}\/$ [Group 13] - [ 6572646300 ] ^http:\/\/([^\x2F]+\/)+SCANNED\/[A-Z]{2,3}[0-9]{4}[A-Z]{6,9}\/$ [Group 14] [ SCANNED/RZ7498WEXEZB ] ^http:\/\/([^\x2F]+\/)+[A-Za-z]{1,4}[0-9]{4,5}[a-zA-Z]{1,5}\/$ [Group 15] [ K44975X ]

#### Rule Vetting

The last step is to check the PCRE's against a corpus of random URL's and see if they appear strict enough in their matching to be used in a production environment. This is critical if you plan to use them for blocking instead of just identification. I can't stress enough how important this phase is; while it's nice to be alerted on access to one of these URL's, it's solid gold if you can prevent attacks and C2 from happening in the first place. Of course, with any blocking action, the caveat is that one wrong block could spell disaster so these need to be as close to perfect as possible.

Ideally, you want to test against a large amount of URL's from your own environment that most closely resemble what traffic your users generate. Unfortunately that's not always possible, or you don't have users, so you need to either build your own corpus or find someone who can test the PCRE's for you.

There isn't much online in the way of random URL lists or logs but I've put together a few possible methods one could try to compile a fairly random set of URL's, and then I'll detail my preferred method.

- Setup a TOR exit node for TCP/80 and just scrape URL's as they traverse.

- Hit up a site like Lenny Zeltser's blocklist page to get a list of other frequently updated blocklists. These are mainly malicious though so not quite random.
- Search Pastebin for "http" and pull pull out URL's. They have a Pastebin Scraping API ($24/yr) to pull down the most recent posts. Sometimes you'll find huge lists with little effort.
- Use some open data sets. Definitely not too random and much more limited in scope.
- Use Twitter Streaming API and filter Tweets for "http".

The Twitter option works nicely and can generate hundreds of thousands of unique URL's per day. Given enough time, you'll have a solid base to test your PCRE's against.

To do this, you need to register an app with Twitter and get your API keys. Once you have those, I've included a Python script, twitter_scraper that you can input them into and run in a continous loop with a one-liner like the below.

while true; do sleep 5; python twitter_scraper.py >> twitter_urls; done

I've also included 2 million URL's on GitHub, which is just under the 25MB file limit compressed. These are ones that I've scraped in the past few days and should help you get started.

Typically I'll check this every so often and filter out things like URL shortening services or other sites that, for one reason or another, have bubbled up to the top of my domain list. This keeps it filled with fairly unique sites and helps improve entropy.

Below is a GIF of the sites streaming by in real time, showing some of the variety.


Once we have our list, we can run pcre_check against the URL's and see how our PCRE's fare.

$ python pcre_check.py -u twitter_urls -p emotet_pcres -s [+] FOUND [+] Count: 1290/2000000 Comment: [Group 13] - [ 6572646300 ] PCRE: ^http:\/\/([^\x2F]+\/)+(?<!appmanager\/)[0-9]{10,11}\/$ [-] MATCH [-] http://db.netkeiba.com/horse/1985105175/ ... http://www.northernminer.com/news/lukas-lundin-copper-commodity-choice/1003786598/ http://www.oita-trinita.co.jp/news/20170532318/ ... http://www.schuh.co.uk/womens/irregular-choice-x-disney-how-do-i-look?-pink-flat-shoes/1364153360/ ... http://www.yutaro-miura.com/info/event/2017/0528100324/ http://yapi.ta2o.net/maseli/2017052901/ [+] FOUND [+] Count: 2595/2000000 Comment: [Group 15] [ K44975X ] PCRE: ^http:\/\/([^\x2F]+\/)+[A-Za-z]{1,4}[0-9]{4,5}[a-zA-Z]{1,5}\/$ [-] MATCH [-] http://epcaf.com/c2805tw/ ... http://hobbyostrov.ru/automodels/electro-

monster-1-10/tra3602g/ ... http://monipla.jp/mfpa/card2017ss/ http://ncode.syosetu.com/N0588Q/ ... http://www.nollieskateboarding.com/fs5050grind/ http://www.profootballweekly.com/2017/05/30/victor-cruz-prepared-to-produce-and-mentor-in-chicago-bears-transitioning-wr-corps/a4613p/

Using the "-s" (show matches) flag in pcre_check will allow you to manually review the false positives. If the sites don't look legitimate or match a little too perfectly, you'll want to do a little manual research to make sure they are in fact FP's and not true positives you didn't know about. I've truncated the results but above shows a few under each to give you an idea of the kind of output I'm looking for to conclude it's not up-to-par.

As you can see, Group 13 and 15 have numerous false-positives. This isn't surprising given Group 13 is simply 10 digits and Group 15 is a small range of alpha, digits, alpha, which continued to repeat itself throughout my analysis.

Additionally, I sent these PCRE's to some fellow miscreant punchers who ran them through over billions of URL's from their environment and received similar output with FP's only for Group 13 and 15.

The last check I'll perform for this set is to remove the trailing forward slash ("/") that was included in the PCRE's. The reason for this is that, while my Emotet seed list all included the forward slash, the URL's I'm scraping may not have it and I just want to try to further identify any potential issues.

$ python pcre_check.py -p emotet_pcres_mod -u twitter_urls

Nadda. Fantastic!

#### Wrapping-up

All in all, 13 total PCRE's make the cut and cover the seen Emotet download URL's. These will provide good historical forensic capability and good passive blocking for future victims of these campaigns.

With that, the below is the final list for publishing and available on GitHub, along with all of the above iterations.

^http:\/\/([^\x2F]+\/)+[a-zA-Z]{1,3}[0-9]{1}[a-zA-Z]{1,3}-[a-zA-Z]{1,2}[0-9]{2,3}-[a-zA-Z]{1,5}\/$ karttoon 31MAY2017 - Emotet download - [ t5wx-x064-mzdb ] ^http:\/\/([^\x2F]+\/)+(INVOICE|ORDER|CUST|Invoice|Cust)-[0-9]{6,7}-[0-9]{4,5}\/$ karttoon 31MAY2017 - Emotet download - [ INVOICE-864339-98261 ] ^http:\/\/([^\x2F]+\/)+dhl\/paket\/com\/pkp\/appmanager\/[0-9]{10}\/$ karttoon 31MAY2017 - Emotet download - [ dhl/paket/com/pkp/appmanager/8376315127 ] ^http:\/\/([^\x2F]+\/)+[A-Z]{2,3}-[0-9]{8}\.dokument\/$ karttoon 31MAY2017 - Emotet download - [ RRT-13279129.dokument ] ^http:\/\/([^\x2F]+\/)+[A-Z]{2,5}(-[0-9]{2})?-[0-9]{5,10}-(document|doc)-May-[0-9]{2}-2017\/$ karttoon 31MAY2017 - Emotet download - [ EDHFR-08-77623-document-May-04-2017 ] ^http:\/\/([^\x2F]+\/)+download[0-9]{4}\/$ karttoon 31MAY2017 - Emotet download - [ download2467 ] ^http:\/\/([^\x2F]+\/)+[a-zA-Z0-9]{1,4}[0-9]{3}[a-zA-Z]{1,5}[0-9]{3}-[a-zA-Z]{1,3}\/$ karttoon 31MAY2017 - Emotet download - [ LUqc663BAyN333-HoO ] ^http:\/\/([^\x2F]+\/)+[A-Z]{4,5}-[A-Z]{1,4}-[0-9]{5}-DE\/$ karttoon 31MAY2017 - Emotet download - [ NUDA-X-52454-DE ] ^http:\/\/([^\x2F]+\/)+(CUST|ORDER|Cust|Rech|Rechnung)(.)?(-Document)?-[A-Z]{1,4}-[0-9]{2,3}-[A-Z]{1,3}[0-9]{4,6}\/$ karttoon 31MAY2017 - Emotet download - [ CUST.-Document-YDI-04-GQ389557 ] ^http:\/\/([^\x2F]+\/)+[a-z0-9]{1,5}-[a-z0-9]{2,4}(-[a-z0-9]{4,5})?-[a-z]{1,4}\.(view|doc)\/$ karttoon 31MAY2017 - Emotet download - [ zp3x-r88-

wuh.view ] ^http:\/\/([^\x2F]+\/)+dhl___status___[0-9]{10}\/$ karttoon 31MAY2017 - Emotet download - [ dhl___status___2668292851 ] ^http:\/\/([^\x2F]+\/)+(ORDER|Rech|CUST|Cust|gescanntes|Scan)(.)?(-Document|-Dokument)?-[0-9]{10,11}\/$ karttoon 31MAY2017 - Emotet download - [ ORDER.-5883789520 ] ^http:\/\/([^\x2F]+\/)+SCANNED\/[A-Z]{2,3}[0-9]{4}[A-Z]{6,9}\/$ karttoon 31MAY2017 - Emotet download [ SCANNED/RZ7498WEXEZB ]

Hopefully this was helpful to some and demonstrated the ease in which these can be created to identify malicious patterns.

The more the merrier in the sharing community!

Ciao!

Older posts...

---