A nickel tour of the ad fraud ecosystem

Ryan Castellucci, Principal Security Researcher, White Ops

Important Jargon

- “Placement” - An available ad location in an individual page view
- “Creative” - The ad content. Usually able to include arbitrary HTML/JavaScript
  - Usually has to be scanned for “badness” and approved before it can run
- “Verification” - Make sure the the parties are getting what they think they’re paying for
- “Traffic” - Page views

WTF is “ad tech”

- Most digital advertising is “programmatic”
- Many players involved
- Much jargon
- Advertisers
- Media Agencies
- DSPs “Demand Side Platforms”
- SSPs “Supply Side Platforms”
- Publishers (aka websites)
- Content Creators

The Players - Advertisers

- Provide the system with money
- Several possible goals
  - “Brand Awareness” - “Hey, Slurm exists! Remember when you get thirsty!”
  - “Lead Generation” - “Give us your email or ‘like’ us on social media!”
  - “Action” - “Click here to buy! Sign up for a free trial! Subscribe to our service!”
- May make a “creative”, but usually outsourced to a Creative Agency
- May dictate tracking and/or verification vendors, usually Media Agency picks

The Players - Media Agencies

- Handles technical details of running campaigns
- Decides how to spend the advertisers money based on their goals
- Selects tracking and verification vendors to be used
- Gets paid a percentage of spend + hourly fees

The Players - Demand Side Platforms

- Interface to buy ad “placements”
- Allows targeting specific “audience segments”
- Usually self-service, some have options for full service
- Many competing companies
- Run “real time bidding” platforms
- Gets paid a percentage of spend

The Players - Supply Side Platforms

- Aggregate publishers
- May also aggregate other SSPs (arbitrage)
- Low end: self-service for small sites
- High end: full service negotiated deals with big sites
- May work with verification and tracking vendors
  - Increase value
  - Decrease reputational risk
  - Manage financial liability (contracts specify a vendor for “billing purposes”)
- Sells via “real time bidding” platforms
- Gets paid a percentage of spend

The Players - Publishers

- Run websites
- Host first and/or third party content
- May pre-sell traffic directly to DSPs, Media Agencies or Advertisers
- May need to buy more traffic to fulfill contracts if they don’t get it organically
- May work with tracking and verification vendors
  - Increase value
  - Decrease reputational risk of purchased traffic
- May revenue share with Content Creators
- Gets paid the biggest share of the spend

The Players - Content Creators

- Host content with Publisher
- Get paid what the publisher says they owe them
- Generally little leverage

The System - Real Time Bidding

- Ad held for SSP for 10s of ms while they await offers from DSPs
- Commonly available information
  - IP address
  - User-Agent string
  - Referer header
  - Publisher, domain, page and app information, as applicable
  - Geolocation data
  - Demographics from tracking companies
  - Placement size

The System - Verification, Tracking & Measurement

- Fraud Detection - is the traffic “valid”/viewed by a human
- Brand Safety - is the ad on a porn site/piracy site/extremist site/etc
- Viewability - did the ad actually get displayed
  - IntersectionObserver is new, and brings some sanity to this
- Tracking - mo’ data, mo’ money
- Measurement - Independent accounting of traffic

Fraud

- Be a “Content Creator”, hire “traffic”, get paid by publisher
- Be a “Publisher”, hire “traffic”, get paid by SSPs
  - “cashout sites” or “ghost sites”
- Be an “SSP”, create “inventory”, get paid by DSPs
  - Ad injection (malware, evil proxies/vpns, dnschanger, etc)
  - Vertically integrated bots that create “traffic” and “inventory” (Methbot)
- Have bots, sell “traffic”
  - 0wn end users
  - Run a “bot farm”
- Have shady site, disguising source of traffic (“traffic laundering”)
- Have site, push affiliate cookies (“cookie stuffing”)
- Have site, run lots of invisible ads (“ad stacking”)

Bot Designs - curl/wget

- A very small shell script
- Figure out the URL that triggers a billing event, hit it
- Great for your IoT botnet
- Very easy to catch by anyone who cares

Bot Designs - Scripts

- Basic web scraper/crawler type code
- Usually written in something like python, node, perl, php, ruby, etc
- Can parse HTML
- Doesn’t execute JavaScript
- Fairly easy to catch by anyone who cares

Bot Designs - Off-the-shelf headless browsers

- Repurposed tools designed for scraping or QA
- Runs without displaying anything
- PhantomJS, SlimerJS, Zombie.js, HtmlUnit, etc.
- Unmodified, detectable with a little effort
- Minor modifications for stealth make detection tricky

Bot Designs - Embedded

- A rendering engine is embedded in another application
- Internet Explorer, Chromium, and Webkit all have supported embedding tools
  - IE WebBrowser control, MSHTML
  - Chromium Embedded Framework
  - WebKit
  - Official support for embedding Gecko was dropped in 2011. Can still be done.
- Usually intended for rendering trusted content
- May have security controls disabled
- Range widely in detection difficulty

Bot Designs - Off-the-shelf automation tools

- Repurposed tools designed for scraping or QA
- Hooks into a real web browser and automates it
- Selenium, Webdriver, and their various wrappers
  - There’s currently a draft W3C spec for webdriver, supposed to set navigator.webdriver = true
- Not usually suitable for compromised end user systems
- Can be difficult to detect

Bot Designs - System Emulators

- Primarily done for bots wanting to run mobile traffic
- Usually combined with off-the-shelf automation tools
- Also done to run “bot farms”
- Tricky to detect

Bot Designs - Custom Browser

- Implement enough of a browser to make verification vendors happy
- Large development effort
- High maintenance
- Deep control of behaviour
- Didn’t expect anyone to actually do this, but we found one

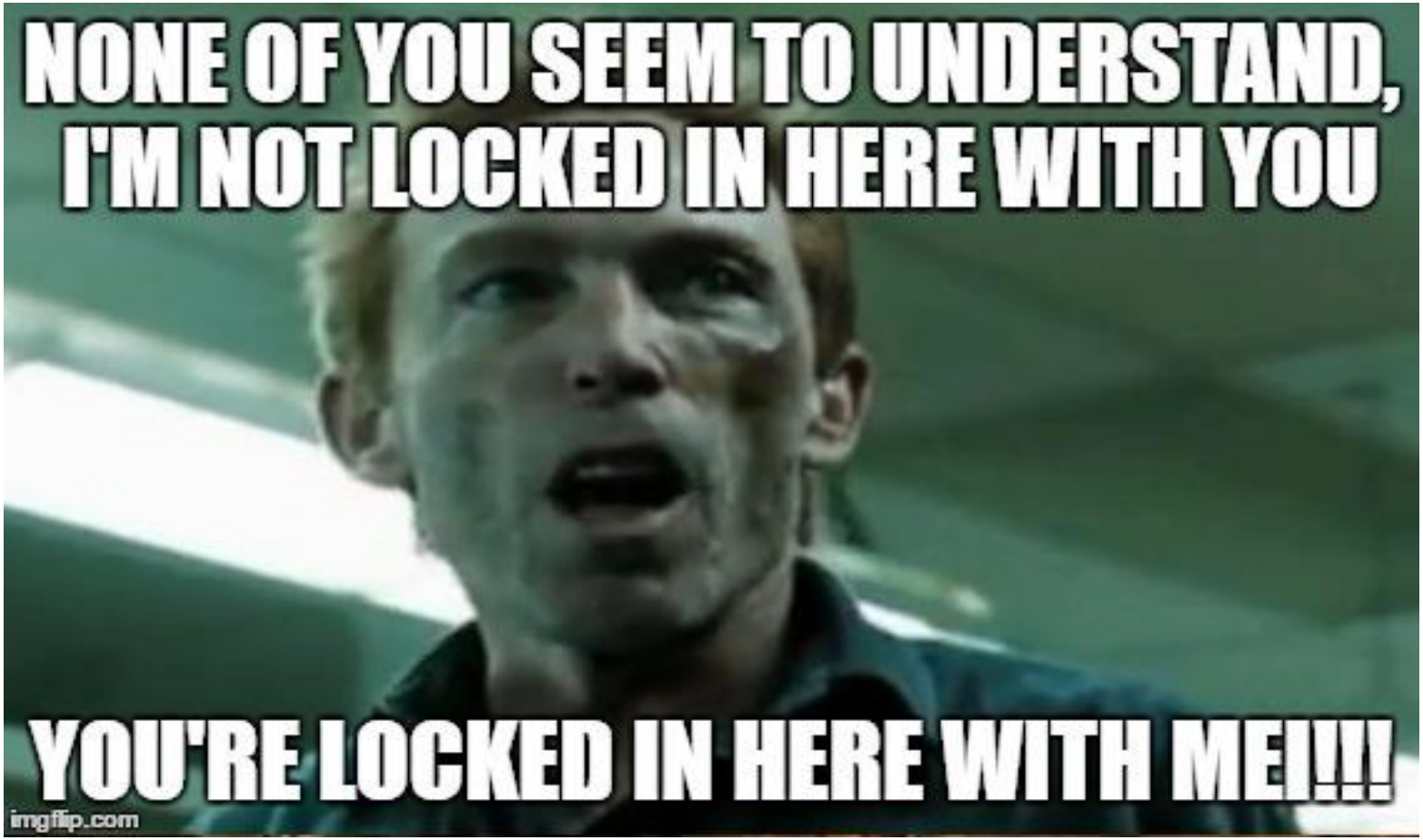
Methbot

- At peak, 300M video ad impressions per day, for millions of dollars
- Hundreds of thousands of IPs falsely registered as US ISPs
  - No, not BGP hijacks, large block allocations and small leased blocks
- Custom HTTP library (buggy)
- DOM support via Cheerio
- CSS support (library unknown)
- Fully custom implementations of many browser APIs
- Flash support via custom NPAPI implementation and Fresh Player
- NodeJS runtime
- A “bot farm” running on dedicated servers
- Extensive fraud detection countermeasures

Methbot - Running third party code

```
result = vm.runInContext(code, this.__MethGlobal, {timeout:EVAL_TIMEOUT});
```

- Essentially, eval
- Allows a substitute global object
- Can apply a timeout to async code


NONE OF YOU SEEM TO UNDERSTAND,
I'M NOT LOCKED IN HERE WITH YOU
YOU'RE LOCKED IN HERE WITH ME!!!!
imgflip.com

Methbot - Dumping code

```
var ary = Object.keys(window), dumpf, dumpt, dumpc;
// grab a random object from the global namespace
var rndObj = window[ary[(Math.random()*ary.length)|0]];
// wrap a hopefully untampered toString function
var str = function(o){return (function(){}).toString.apply(o)};
// try to dump some code
try{ dumpf = str(rndObj)              }catch(e){}
try{ dumpt = str(rndObj.toString)     }catch(e){}
try{ dumpc = str(rndObj.constructor) }catch(e){}
```

Methbot - Don’t dump me bro

```
text = text.split('function() {}.toString.apply(')
           .join('window.__MethFakedFuncToString(');
text = text.split('function(){}.toString.apply(')
           .join('window.__MethFakedFuncToString(');
text = text.split('{}.toString.apply(')
           .join('window.__MethFakedToString(');
```

Methbot - __MethFakedToString

```
__MethFakedToString = function(el){
    try {
        if (el.hasOwnProperty('toString'))
            return el.toString()
    } catch (el) {}
    return {}.toString.apply(el)
}
```

Methbot - __MethFakedFuncToString

```
__MethFakedFuncToString = function(el){
    try {
        if (el.hasOwnProperty('toString'))
            return el.toString()
    } catch (e) {}
    var t = null;
    t = function() {}.toString.apply(el)
    return t;
}
```

Methbot - Blocking our flash

```
// return loadLocalFile(this.link, this.callback,
// '../for_whiteops/load.src.4.16.6.js')
//}
// wo flash
if (this.link.indexOf('viz11.swf') !== -1) {
var res = {
  url: this.link, statusCode: 200, status: '200 OK',
  rawHeaders:'HTTP/1.1 200 OK\nServer: nginx/1.4.6 (Ubuntu)\n',
  headers: {}, $:cheerio.load(''), body:new Buffer('')
};
return this.callback(false, res)
}
```

Methbot - Typos

```
Navigator.prototype = {
/*...*/
  appCodeName: {value:"Mozzila"},
/*...*/
}
// from plugin spoofing code
var fname = o === "W" ? "pepflashplayer.dll " : "PepperFlashPlayer.plugin";
var pl_swf = {
  description:"Shockwave Flash 23.0 r0",
  filename:fname,
  name:"Shockwave Flash"
};
```

Methbot - IP Registration Forgery

```
inetnum:        196.62.0.0 - 196.62.31.255
netname:        COMCAST-CABLE
descr:          Comcast Cable Communications, Inc
country:        US
admin-c:        IP9-AFRINIC
tech-c:         IP9-AFRINIC
status:         ASSIGNED PA
mnt-by:         IP-ADMIN
mnt-lower:      IP-ADMIN
mnt-domains:    IP-ADMIN
mnt-routes:     IP-ADMIN
changed:        adw0rd.yandex.ru@gmail.com 20151014
source:         AFRINIC
parent:         196.62.0.0 - 196.62.255.255
```

```
person:         IP Admin
address:        IP Admin
phone:          +2482534202
e-mail:         adw0rd.yandex.ru@gmail.com
nic-hdl:        IP9-AFRINIC
changed:        adw0rd.yandex.ru@gmail.com 20151014
source:         AFRINIC
```

Methbot - IP Registration Forgery

```
inetnum:        196.62.32.0 - 196.62.63.255
netname:        TIME-WARNER
descr:          Time Warner Cable Inc.
country:        US
admin-c:        IP9-AFRINIC
tech-c:         IP9-AFRINIC
status:         ASSIGNED PA
mnt-by:         IP-ADMIN
mnt-lower:      IP-ADMIN
mnt-domains:    IP-ADMIN
mnt-routes:     IP-ADMIN
source:         AFRINIC # Filtered
parent:         196.62.0.0 - 196.62.255.255
```

Methbot - IP Registration Forgery

```
inetnum:        196.62.64.0 - 196.62.95.255
netname:        VERIZON
descr:          Verizon Trademark Services LLC
country:        US
admin-c:        IP9-AFRINIC
tech-c:         IP9-AFRINIC
status:         ASSIGNED PA
mnt-by:         IP-ADMIN
mnt-lower:      IP-ADMIN
mnt-domains:    IP-ADMIN
mnt-routes:     IP-ADMIN
source:         AFRINIC # Filtered
parent:         196.62.0.0 - 196.62.255.255
```

Methbot - IP Registration Forgery

```
inetnum:        196.62.96.0 - 196.62.127.255
netname:        ATT
descr:          AT&T Services, Inc.
country:        US
admin-c:        IP9-AFRINIC
tech-c:         IP9-AFRINIC
status:         ASSIGNED PA
mnt-by:         IP-ADMIN
mnt-lower:      IP-ADMIN
mnt-domains:    IP-ADMIN
mnt-routes:     IP-ADMIN
source:         AFRINIC # Filtered
parent:         196.62.0.0 - 196.62.255.255
```

Methbot - IP Registration Forgery

```
inetnum:        196.62.128.0 - 196.62.159.255
netname:        COX
descr:          Cox Communications Inc
country:        US
admin-c:        IP9-AFRINIC
tech-c:         IP9-AFRINIC
status:         ASSIGNED PA
mnt-by:         IP-ADMIN
mnt-lower:      IP-ADMIN
mnt-domains:    IP-ADMIN
mnt-routes:     IP-ADMIN
source:         AFRINIC # Filtered
parent:         196.62.0.0 - 196.62.255.255
```

Methbot - IP Registration Forgery

```
inetnum:        196.62.160.0 - 196.62.191.255
netname:        CHARTER
descr:          Charter Communications Operating, LLC
country:        US
admin-c:        IP9-AFRINIC
tech-c:         IP9-AFRINIC
status:         ASSIGNED PA
mnt-by:         IP-ADMIN
mnt-lower:      IP-ADMIN
mnt-domains:    IP-ADMIN
mnt-routes:     IP-ADMIN
source:         AFRINIC # Filtered
parent:         196.62.0.0 - 196.62.255.255
```

Methbot - IP Registration Forgery

```
inetnum:        196.62.192.0 - 196.62.223.255
netname:        Cequel
descr:          Cequel Communications Holdings
country:        US
admin-c:        IP9-AFRINIC
tech-c:         IP9-AFRINIC
status:         ASSIGNED PA
mnt-by:         IP-ADMIN
mnt-lower:      IP-ADMIN
mnt-domains:    IP-ADMIN
mnt-routes:     IP-ADMIN
source:         AFRINIC # Filtered
parent:         196.62.0.0 - 196.62.255.255
```

Methbot - IP Registration Forgery

```
inetnum:        196.62.224.0 - 196.62.255.255
netname:        CenturyLink
descr:          CenturyLink, Inc.
country:        US
admin-c:        IP9-AFRINIC
tech-c:         IP9-AFRINIC
status:         ASSIGNED PA
mnt-by:         IP-ADMIN
mnt-lower:      IP-ADMIN
mnt-domains:    IP-ADMIN
mnt-routes:     IP-ADMIN
source:         AFRINIC # Filtered
parent:         196.62.0.0 - 196.62.255.255
```

Methbot - IP Registration Forgery

```
% Abuse contact for '161.8.192.0 - 161.8.223.255' is 'stepanenko.aa@mmk.ru'

inetnum:        161.8.192.0 - 161.8.223.255
netname:        Verizon_Trademark_Services_LLC-19
descr:          Verizon Trademark Services LLC
country:        US
admin-c:        SOV68-RIPE
tech-c:         SOV68-RIPE
status:         LEGACY
mnt-by:         MMKMGN-MNT
mnt-by:         NetBC
created:        2015-10-13T14:47:56Z
last-modified:  2015-10-13T14:47:56Z
source:         RIPE

person:         NetBComm LLC
address:        USA, Texas, Dallas , Verizon
Trademark Services LLC
phone:          +12191278854
nic-hdl:        SOV68-RIPE
mnt-by:         NetBC
created:        2015-07-20T07:15:59Z
last-modified:  2015-12-25T08:57:55Z
source:         RIPE # Filtered
```

Other JavaScript Dumping Countermeasures

```
function toString() {
  // An if-else chain is used here because a "switch" block or an Object lookup
  // would coerce these functions into strings.
  if (this === _functionToStringShim) {
    var target = _functionToStringOrig;
  } else if (this === _alertShim) {
    target = _alertOrig;
  } else if (this === _confirmShim) {
    target = _confirmOrig;
  /* This code has been modified from its original version. It has been formatted to fit this slide. */
  } else if (this == _getCurrentPositionShim) {
    target = _getCurrentPositionOrig;
  } else if (this === _onmessageDelegate && _onmessageFormatted != null) {
    return _onmessageFormatted;
  } else {
    target = this;
  }
  return sandbox('Function', 'toString')(target);
}
```

Other JavaScript Dumping Countermeasures

```
var _fpts = Function.prototype.toString; // save reference
Function.prototype.toString = (function(){ // setup spoofing
  var fakeToString = function toString() {
    if (this === fakeToString) {
      return _fpts.apply(_fpts, arguments);
      /* more evil spoofing logic goes here */
    } else { return _fpts.apply(this, arguments); }
  };
  return fakeToString;
})();
```

Bot Detection - Blacklists

- IP addresses (datacenters, open proxies, etc)
- User-Agent strings
- App IDs
- Domains
- IAB/ABC International Spiders and Bots List
  - Paid subscription
  - Complicated to implement (we're releasing ours as open source soon, https://github.com/whiteops-dot-com/spidersandbots)
  - Intended mainly to filter "legitimate" bots

Bot Detection - Consistency

- Do HTTP header values (User-Agent, Language) match JavaScript data?
- Do the plugins have the right file extensions for the claimed OS?
- Does the OS reported by Flash match the User-Agent?
- Dozens of other hints about what the browser and operating system are
- Bots often can’t keep their story straight

Bot Detection - Statistical Anomalies

- Skewed OS, Browser, Device, Resolution, etc distribution
- Too much traffic from too few IPs
- IPs or “users” visiting too many or too few domains or pages
- IPs or “users” visiting weird combinations of pages
- Strange timing of traffic patterns
- Time spent on pages too high/low/regular
- Engagement metrics too high/low

Bot Detection - Specific bots/tools

- navigator.webdriver is true
- document.documentElement.getAttribute("webdriver") is true
- window.callPhantom or window._phantom exists
- window.alert overridden
- console.log overridden
- Weird HTTP headers such as “Content-Suport” present
- HTTP header “Cache-Control” contains “:” (Methbot)
- Same Origin Policy or other security controls disabled

Bot Detection - Flash

- Compare “capabilities” data with JavaScript data
- Unexpected or known bad values in “capabilities” string
- Rendering throttle events (caused by not being on screen)
- No hardware video acceleration (VM)
- No microphone device (VM)

Questions?

Twitter: @ryancdotorg

Github: https://github.com/ryancdotorg

Personal blog I post on maybe once a quarter: https://rya.nc/

These slides (give it a few hours): https://rya.nc/shmoo2017

Methbot Whitepaper: https://w-ops.com/methbot_wp

Methbot IP list: https://www.whiteops.com/methbot/IPs-CIDR.txt