28
JUN
2018

BLUE TEAM, HOW-TO, PHISHING ANTI-PHISING, BEST PRACTICES, BLUE TEAM, DKIM, DMARC, EMAIL, FILTERING, INCIDENT RESPONSE, IR, MARKETING, PHISHING, RECONNAISSANCE, RFC 4408, SENDER POLICY FRAMEWORK, SPAM, SPF

Offensive SPF: How to Automate Anti-Phishing Reconnaissance Using Sender Policy Framework

Kent Ickler//

FOLLOW US

TL;DR: *This post describes the process of building an active system to automatically recon SPF violations.*

Disclaimer:

LOOKING FOR SOMETHING?

SUBSCRIBE TO THE BHISBLOG

Don't get left in the dark! Enter your email address and every time a post goes live you'll get instant notification! We'll also add you to our webcast list, so you won't miss our occasional emails about upcoming events! (We promise, we're not spammy!)

Email Address

Subscribe

There are parts of this build that might not be legal in your area. Use in the wild at your own risk. Discuss with your peeps before implementing. BHIS @Krelkci are not liable for your actions.

Background:

In our previous blog post about configuring SPF, I didn't elaborate on the awesomeness of the exist and reason mechanics. What little people, outside of SPF experts, know is that you can build a system of response automation around the use of these two mechanics. Like to read? Syntax: RFC 4408 http://www.openspf.org/RFC_4408

The exists mechanic will force a (compliant) receiving mail server to check if a specific A DNS record exists for a specific domain. While that seems interesting and all, what perhaps is more important is the use of SPF macros within the exists mechanic. It essentially allows you to pass information about the originating SMTP server from the receiving SMTP server to wherever the domain owner of the domain in the envelope's FROM field determines.

How you say?

Let's look at this SPF record:

```
v=spfc1 include:mail.youdomain.com -exists:{d}.AutoRecon.yourdomain.com -all
```

The receiving SMTP server does the following actions:

BROWSE BY CATEGORY

Select Category

RECENT POSTS

Webcast: Windows logging, Sysmon, and ELK

Click on the timecodes to jump to that part of the

How to Hack Hardware using UART

Raymond Felch // Preface: I began my exploration of

Webcast: Implementing Sysmon and Applocker

Click on the timecodes to jump to that part of the

- Receive from originating mail server where the FROM field = domain
- Check SPF record for mail.yourdomain.com, of origin server is found= Good, else = move on.
- Check if an A DNS record exists for [ORIGINATING.MAIL.SERVER.NAME].autorecon.yourdomain.com
- Kill everything else (-all)

Here are the key points. If mail is delivered from a server that doesn't exist within the SPF headers of mail.yourdomain.com, the receiving mail server is going to attempt to check an alias record for a dynamic hostname that is built on the fly. All you have to do now is build a DNS server configured to accept DNS queries for .autorecon.yourdomain.com. and provide all queries to an auto-recon system ,and tell your global DNS provider that autorecon.yourdomain.com is authoritatively answered by your auto-recon service. Let's do it.

On the AutoRecon Service

- Bind configured to accept queries for AutoSPF.yourdomain.com
- SSMTP configured to send mail

Get the files:

```
cd/opt/

git clone https://github.com/Relkci/AutoSPFRecon

apt-get install bind9

apt-get install logtail

apt-get install python-setuptools

easy_install clic

easy_install shodan
```

Setup your BIND9 Domain -named.conf

```
nano /etc/bind/named.conf

zone "autorecon.YOURDOMAIN.com" {
type master;
notify no;
file "/etc/bind/AutoRecon.yourdomain.com";
};
```

Setup your BIND9 Domain – zone file

```
nano /etc/bind/autospf.yourdomain.tld
$TTL 3D
@       IN      SOA     autorecon.ns.yourdomain.com. admin@yourdomain.com (
199802151       ; serial, todays date + todays serial #
21600              ; refresh, seconds
3600              ; retry, seconds
604800              ; expire, seconds
```

ARCHIVES

Select Month

```
30 )            ; minimum, seconds
;
NS      ns              ; Inet Address of name server
;
localhost       A       127.0.0.1
ns      A       IP-OF-AutoRecon
```

Restart Bind

```
Service bind9 restart

Service bind9 status
```

Configure Bind to log DNS queries to /var/log/syslog:

```
#below command toggles query logging, be sure it is enabled

rdnc querylog

#confirm it is turned on with

tail -n 2 /var/log/syslog
```

Setup your Domain DNS records

CAUTION Setting the SPF RECORD AS BELOW WILL TELL ALL MAIL SERVERS TO REJECT YOUR EMAIL**

You can use ?exists:autospf.yourdomain.tld mechanic which will not immediately reject email. Be sure you retain the proper parts of SPF so that you

do not reject all email. The below example would be appropriate for a domain that should never send email.

See our blog post on SPF Records to create a proper SPF record for your organization.

On your TLD nameserver:
Type: A Host: autorecon.ns.yourdomain.com Value: IP-OF-AutoRecon
Type: NS Host: autorecon.yourdomain.com Value: autorecon.ns.yourdomain.com
Type: TXT Host: @ Value: “v=spf1 -exists:%{i}.autorecon.yourdomain.com -all”

Putting it all together:

When a mail server receives email and the originating mail server reviews the SPF record and finds it cannot find the mail server in an include: or other mail record, it will continue until it finds the exists:%{i}.autorecon.yourdomain.com which will instruct it to replace the %{i} with the IP of the server originating the email. The server will lookup the NS record for autorecon.yourdomain.com and find that it is the autorecon.yourdomain.com service. It will query {IP}.autorecon.yourdomain.com and will not receive a valid DNS response. The Bind server on autorecon.yourdomain.com however will have logged the query in /var/log/syslog.

The AutoReconSPF.sh script reads the syslog for those queries, runs a shodan query, and then delivers the results to an email address in question.

The AutoReconSPF.sh script can be configured to run every few minutes with crontabs.

What Else Can it do:

This proof of concept script sets the framework in a compartmentalized and easy to edit way. You can add your own script actions such as NMAP scans, IR events, or maybe even link it back to Fail2Ban or IPTable black-lists.

Expand. NMap, Fail2Ban, IPTables, Incident Response. Automate Lights Out.

Someone attempts to phish your staff with an email forged to be from your domain. Since your SPF records fail to authorize the originating mail server, your AutoSPFRecon system gets alerted and triggers an email, Fail2Ban blockade, and immediately the phishing server's visibility into your infrastructure goes immediately dark.

Running AutoReconSPF.sh

In this test, I have sent an email forged with a domain that had the AutoReconSPF SPF records. The email was sent from a Digital Ocean droplet at 206.189.xxx.xxx. The receiving mail server sends a query to autospf.bhis.io and

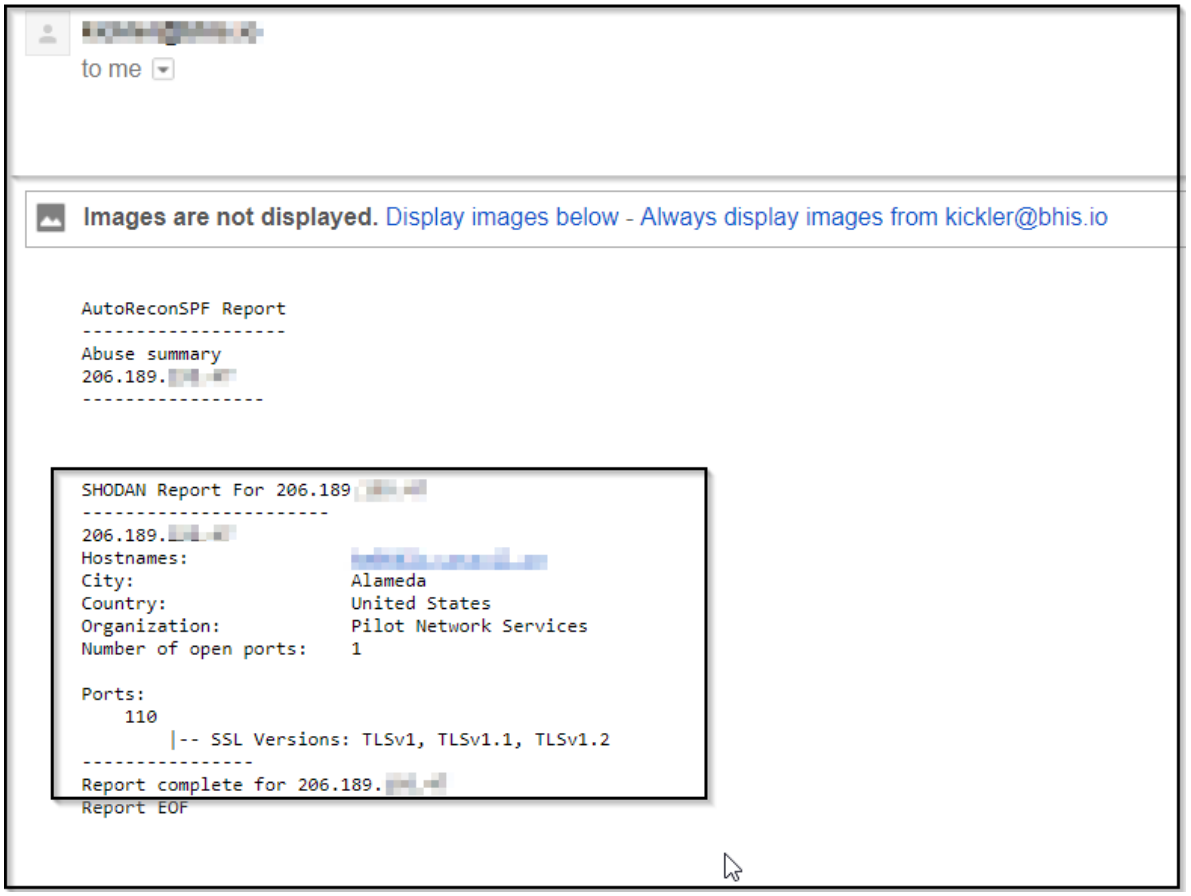
the log entry is created. AutoReconSPF.sh identifies the offending mail server's IP to shodan and sends the results to me in email. Awesome.

```
: /opt/AutoSPFRecon
/opt/AutoSPFRecon# ./AutoReconSPF.sh
Running AutoReconSPF
Initializing Shodan API
Successfully initialized

Running Report for 206.189.

Emailing
Cleanup
Done
:/opt/AutoSPFRecon#
```

Resulting Email Delivered:



Links:

GitHub: https://github.com/Relkci/AutoSPFRecon

RFC: SPF that includes exists: mechanic http://www.openspf.org/RFC_4408

BHIS SPF for the Masses Blog Post: https://www.blackhillsinfosec.com/how-to-configure-spfv1-explained-for-the-masses/

Join the BHIS Blog Mailing List – get notified when we post new blogs, webcasts, and podcasts.

Join 1,544 other subscribers

Email Address

Subscribe

Kent Ickler//

TL;DR: *This post describes the process of building an active system to automatically recon SPF violations.*

Disclaimer:

There are parts of this build that might not be legal in your area. Use in the wild at your own risk. Discuss with your peeps before implementing. BHIS @Krelkci are not liable for your actions.

Background:

In our previous blog post about configuring SPF, I didn't elaborate on the awesomeness of the exist and reason mechanics. What little people, outside of SPF experts, know is that you can build a system of response automation around the use of these two mechanics. Like to read? Syntax: RFC 4408 http://www.openspf.org/RFC_4408

The exists mechanic will force a (compliant) receiving mail server to check if a specific A DNS record exists for a specific domain. While that seems interesting and all, what perhaps is more important is the use of SPF macros within the exists mechanic. It essentially allows you to pass information about the originating SMTP server from the receiving SMTP server to wherever the domain owner of the domain in the envelope's FROM field determines.

How you say?

Let's look at this SPF record:

```
v=spfc1 include:mail.youdomain.com -exists:{d}.AutoRecon.yourdomain.com -all
```

The receiving SMTP server does the following actions:

- Receive from originating mail server where the FROM field = domain
- Check SPF record for mail.yourdomain.com, of origin server is found= Good, else = move on.
- Check if an A DNS record exists for [ORIGINATING.MAIL.SERVER.NAME].autorecon.yourdomain.com
- Kill everything else (-all)

Here are the key points. If mail is delivered from a server that doesn't exist within the SPF headers of mail.yourdomain.com, the receiving mail server is going to attempt to check an alias record for a dynamic hostname that is built on the fly. All you have to do now is build a DNS server configured to accept DNS queries for .autorecon.yourdomain.com. and provide all queries to an auto-recon system ,and tell your global DNS provider that autorecon.yourdomain.com is authoritatively answered by your auto-recon service. Let's do it.

On the AutoRecon Service

- Bind configured to accept queries for AutoSPF.yourdomain.com
- SSMTP configured to send mail

Get the files:

```
cd/opt/

git clone https://github.com/Relkci/AutoSPFRecon

apt-get install bind9

apt-get install logtail

apt-get install python-setuptools

easy_install click

easy_install shodan
```

Setup your BIND9 Domain -named.conf

```
nano /etc/bind/named.conf

zone "autorecon.YOURDOMAIN.com" {
type master;
notify no;
file "/etc/bind/AutoRecon.yourdomain.com";
};
```

Setup your BIND9 Domain – zone file

```
nano /etc/bind/autospf.yourdomain.tld
$TTL 3D
@       IN      SOA     autorecon.ns.yourdomain.com. admin@yourdomain.com (
199802151       ; serial, todays date + todays serial #
21600              ; refresh, seconds
3600              ; retry, seconds
```

```
604800              ; expire, seconds
30 )            ; minimum, seconds
;
NS      ns              ; Inet Address of name server
;
localhost       A       127.0.0.1
ns      A       IP-OF-AutoRecon
```

Restart Bind

```
Service bind9 restart

Service bind9 status
```

Configure Bind to log DNS queries to /var/log/syslog:

```
#below command toggles query logging, be sure it is enabled

rdnc querylog

#confirm it is turned on with

tail -n 2 /var/log/syslog
```

Setup your Domain DNS records

CAUTION Setting the SPF RECORD AS BELOW WILL TELL ALL MAIL SERVERS TO REJECT YOUR EMAIL**

You can use ?exists:autospf.yourdomain.tld mechanic which will not immediately reject email. Be sure you retain the proper parts of SPF so that you do not reject all email. The below example would be appropriate for a domain that should never send email.

See our blog post on SPF Records to create a proper SPF record for your organization.

On your TLD nameserver:
Type: A Host: autorecon.ns.yourdomain.com Value: IP-OF-AutoRecon
Type: NS Host: autorecon.yourdomain.com Value: autorecon.ns.yourdomain.com
Type: TXT Host: @ Value: “v=spf1 -exists:%{i}.autorecon.yourdomain.com -all”

Putting it all together:

When a mail server receives email and the originating mail server reviews the SPF record and finds it cannot find the mail server in an include: or other mail record, it will continue until it finds the exists:%{i}.autorecon.yourdomain.com which will instruct it to replace the %{i} with the IP of the server originating the email. The server will lookup the NS record for autorecon.yourdomain.com and find that it is the autorecon.yourdomain.com service. It will query

{IP}.autorecon.yourdomain.com and will not receive a valid DNS response. The Bind server on autorecon.yourdomain.com however will have logged the query in /var/log/syslog.

The AutoReconSPF.sh script reads the syslog for those queries, runs a shodan query, and then delivers the results to an email address in question.

The AutoReconSPF.sh script can be configured to run every few minutes with crontabs.

What Else Can it do:

This proof of concept script sets the framework in a compartmentalized and easy to edit way. You can add your own script actions such as NMAP scans, IR events, or maybe even link it back to Fail2Ban or IPTable black-lists.

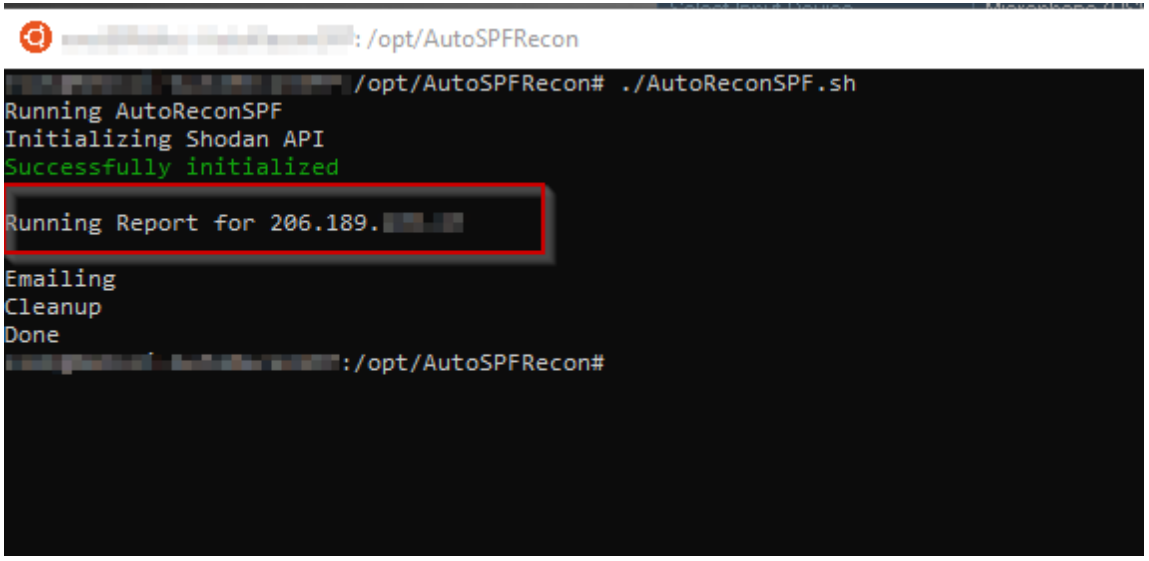
Expand. NMap, Fail2Ban, IPTables, Incident Response. Automate Lights Out.

Someone attempts to phish your staff with an email forged to be from your domain. Since your SPF records fail to authorize the originating mail server, your AutoSPFRecon system gets alerted and triggers an email, Fail2Ban

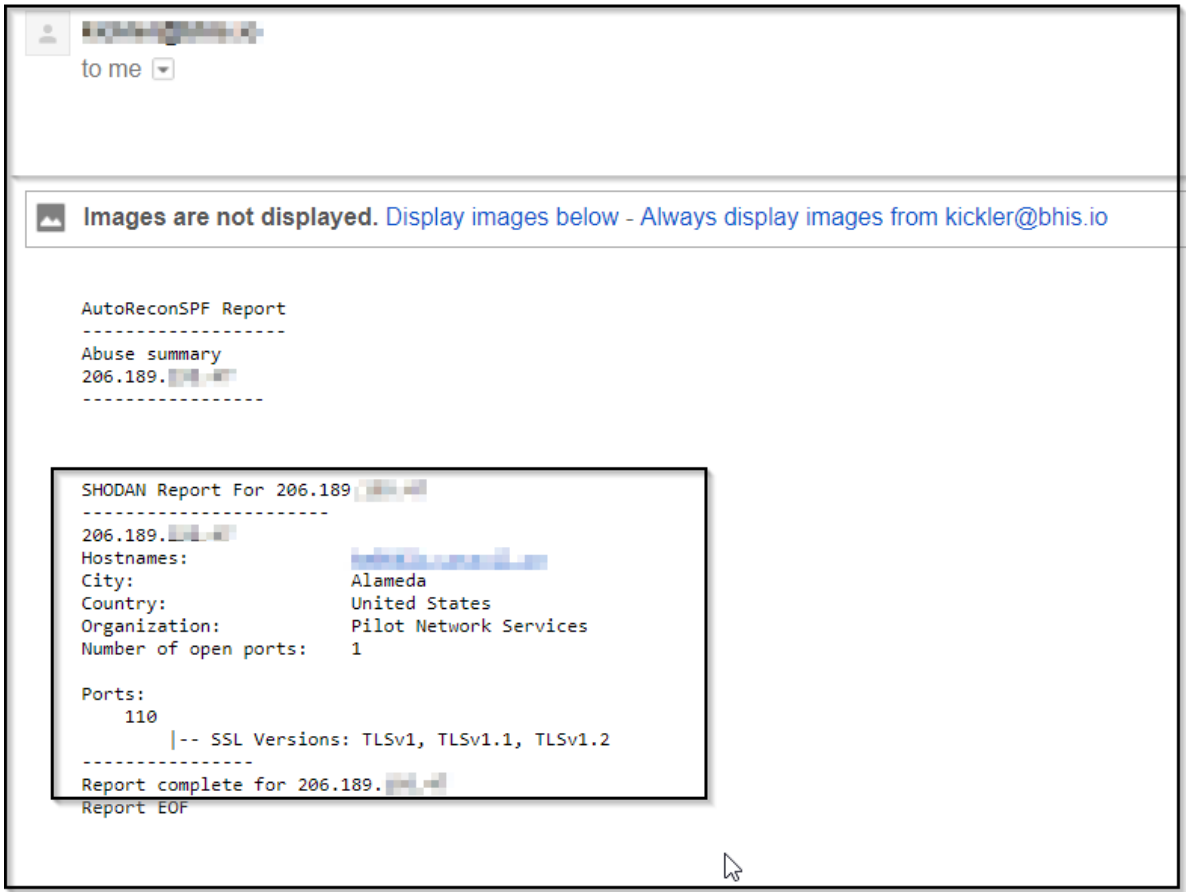
blockade, and immediately the phishing server's visibility into your infrastructure goes immediately dark.

Running AutoReconSPF.sh

In this test, I have sent an email forged with a domain that had the AutoReconSPF SPF records. The email was sent from a Digital Ocean droplet at 206.189.xxx.xxx. The receiving mail server sends a query to autospf.bhis.io and the log entry is created. AutoReconSPF.sh identifies the offending mail server's IP to shodan and sends the results to me in email. Awesome.



Resulting Email Delivered:



Links:

GitHub: https://github.com/Relkci/AutoSPFRecon

RFC: SPF that includes exists: mechanic http://www.openspf.org/RFC_4408

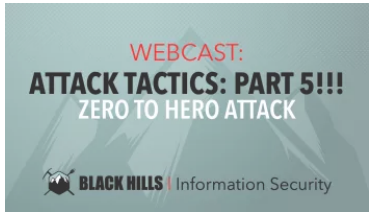
BHIS SPF for the Masses Blog Post: https://www.blackhillsinfosec.com/how-to-configure-spfv1-explained-for-the-masses/

Share this:

Related

How to Configure SPFv1: Explained for the Masses

May 29, 2018
In "Blue Team"



Webcast: Attack Tactics 5 - Zero to Hero Attack

May 1, 2019
In "C2"

Information Security Glossary - v2

Original by Bob Covello, CISSP / Modified with permission by BHIS // Note: This glossary was started to answer questions related to

March 28, 2016
In "InfoSec 101"

Subscribing to Our Podcast

WEBCAST: Hacker Tools, Compliments of Microsoft

BLACK HILLS INFORMATION SECURITY

115 W. Hudson St. Spearfish, SD 57783 | 701-484-BHIS

LINKS

SEARCH THE SITE