SLASHNEXT

USER GUIDE V1.1.0

SlashNext Threat Intelligence Integration Guide MineMeld

TABLE OF CONTENTS

1 | INTRODUCTION

This document provides a detailed tutorial on integrating SlashNext's threat intelligence feed into PaloAlto Network's MineMeld by creating custom miner, aggregator, and output prototypes and nodes.

> ⓘ **Warning**
>
> It is important to note that these prototypes cannot be edited after their creation so if something goes wrong with the settings the corresponding prototype shall be deleted and re-created.

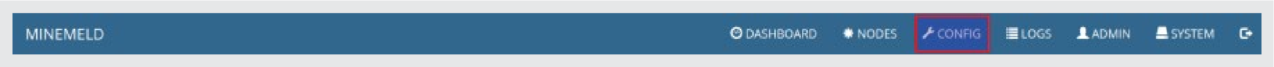
The integration consists the following 6 steps:

1. Create custom SlashNext miner prototypes suitable for JSON API endpoint
2. Create custom SlashNext miner nodes using the new SlashNext miner prototypes
3. Create a custom SlashNext processor prototype
4. Create a custom SlashNext processor node based on the new SlashNext processor prototype
5. Create a custom SlashNext output prototype using the new SlashNext output prototype
6. Create a custom SlashNext output node based on the new SlashNext output prototype

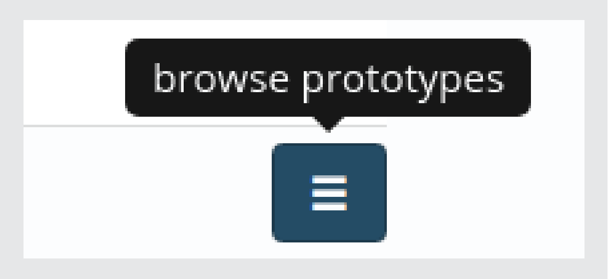
2 | MINEMELD CONFIGURATION

2.1 |CREATING NEW SLASHNEXT MINER PROTOTYPES

The first step is to create custom miner prototypes which will later be used to create SlashNext custom miner nodes. These prototype define the external feed location and other parameters for the firewall to read it as an external dynamic list (EDL).
After logging into MineMeld, click the **CONFIG** menu-bar option to see the currently configured items.

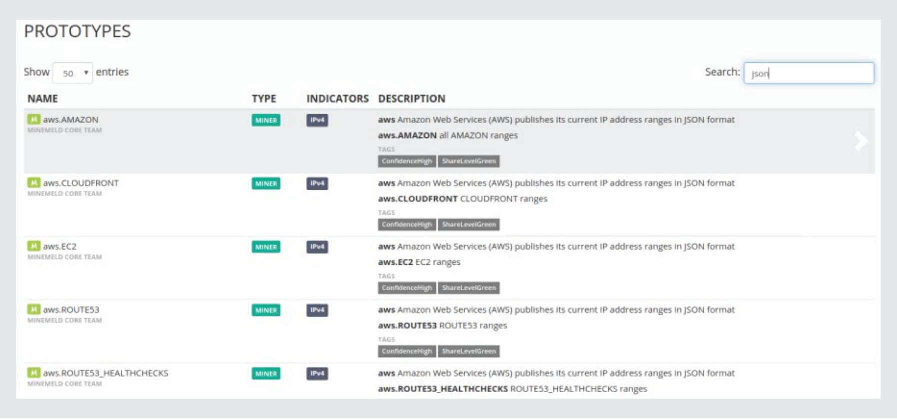


Next, in the lower right-hand portion of the web-page, click on the hamburger icon to browse prototypes.



Search for a prototype miner whose type matches your thread feed source. In this case the miner needs to work with a thread feed having response of JSON type so search for "json". In this document, the aws.Amazon prototype is used to create a custom SlashNext miner prototype.

Search for a prototype miner whose type matches your thread feed source. In this case the miner needs to work with a thread feed having response of JSON type so search for "json". In this document, the aws.Amazon prototype is used to create a custom SlashNext miner prototype.

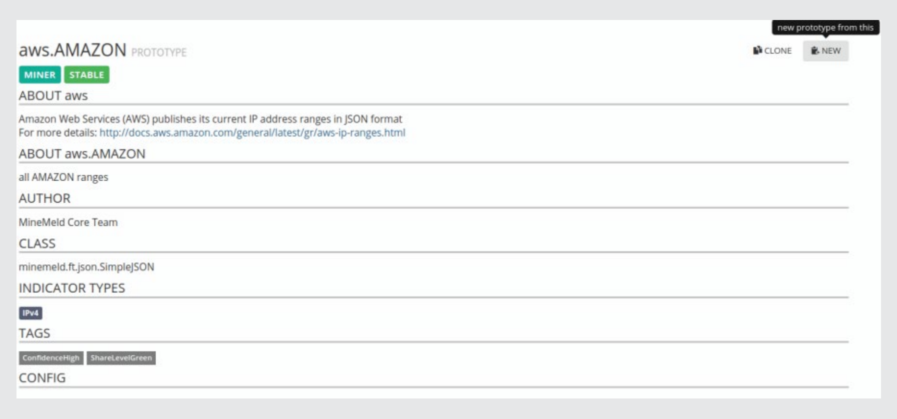


PROTOTYPES

Show 50 entries Search: json

NAME	TYPE	INDICATORS	DESCRIPTION
aws.AMAZON MINEMELD CORE TEAM	MINER	IPv4	**aws** Amazon Web Services (AWS) publishes its current IP address ranges in JSON format **aws.AMAZON** all AMAZON ranges TAGS ConfidenceHigh ShareLevelGreen
aws.CLOUDFRONT MINEMELD CORE TEAM	MINER	IPv4	**aws** Amazon Web Services (AWS) publishes its current IP address ranges in JSON format **aws.CLOUDFRONT** CLOUDFRONT ranges TAGS ConfidenceHigh ShareLevelGreen
aws.EC2 MINEMELD CORE TEAM	MINER	IPv4	**aws** Amazon Web Services (AWS) publishes its current IP address ranges in JSON format **aws.EC2** EC2 ranges TAGS ConfidenceHigh ShareLevelGreen
aws.ROUTE53 MINEMELD CORE TEAM	MINER	IPv4	**aws** Amazon Web Services (AWS) publishes its current IP address ranges in JSON format **aws.ROUTE53** ROUTE53 ranges TAGS ConfidenceHigh ShareLevelGreen
aws.ROUTE53_HEALTHCHECKS MINEMELD CORE TEAM	MINER	IPv4	**aws** Amazon Web Services (AWS) publishes its current IP address ranges in JSON format **aws.ROUTE53_HEALTHCHECKS** ROUTE53_HEALTHCHECKS ranges

Click on the **aws.Amazon** prototype to see the details. Then click on New to create a **new** prototype based on this specific miner.



aws.AMAZON PROTOTYPE

MINER STABLE

ABOUT aws

Amazon Web Services (AWS) publishes its current IP address ranges in JSON format
For more details: http://docs.aws.amazon.com/general/latest/gr/aws-ip-ranges.html

ABOUT aws.AMAZON

all AMAZON ranges

AUTHOR

MineMeld Core Team

CLASS

minemeld.ft.json.SimpleJSON

INDICATOR TYPES

IPv4

TAGS

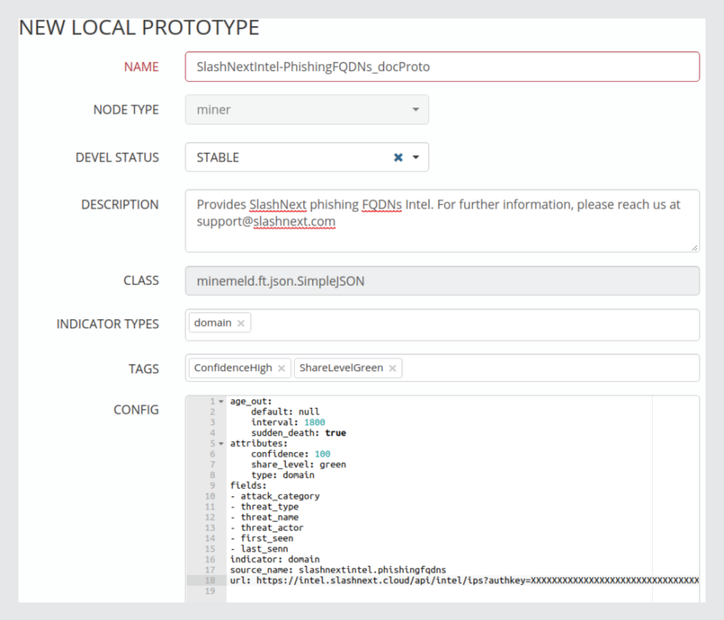
ConfidenceHigh ShareLevelGreen

CONFIG

CONFIG areas as shown below. The **url** field in the **CONFIG** is set to

https://intel.slashnext.cloud/api/intel/domains?authkey=XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX&format=json

which points to the SlashNext thread feed API of JSON type response.



NEW LOCAL PROTOTYPE

NAME: SlashNextIntel-PhishingFQDNs_docProto

NODE TYPE: miner

DEVEL STATUS: STABLE

DESCRIPTION: Provides SlashNext phishing FQDNs Intel. For further information, please reach us at support@slashnext.com

CLASS: minemeld.ft.json.SimpleJSON

INDICATOR TYPES: domain ×

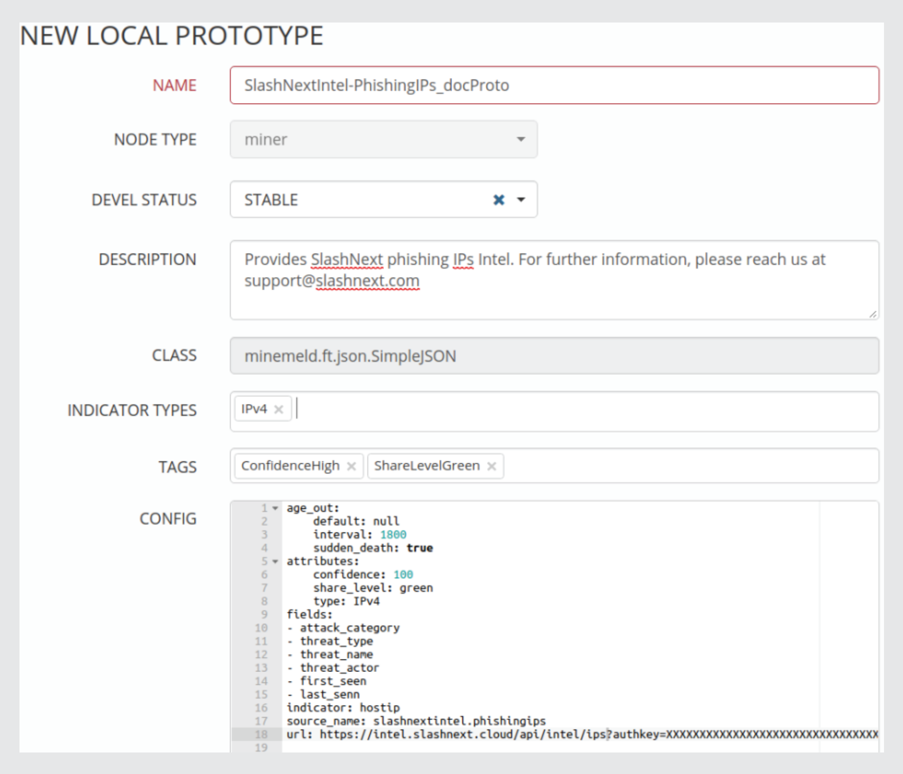
TAGS: ConfidenceHigh × ShareLevelGreen ×

CONFIG:

```
age_out:
    default: null
    interval: 1800
    sudden_death: true
attributes:
    confidence: 100
    share_level: green
    type: domain
fields:
- attack_category
- threat_type
- threat_name
- threat_actor
- first_seen
- last_senn
indicator: domain
source_name: slashnextintel.phishingfqdns
url: https://intel.slashnext.cloud/api/intel/ips?authkey=XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

Click **OK** to save the new prototype. Similarly, repeat the above mentioned steps to create two other prototypes for SlashNext Phishing IPs and Wildcard URLs respectively.

The **CONFIG** settings of each prototype and their respective endpoint URLs are given below in the same order.



NEW LOCAL PROTOTYPE

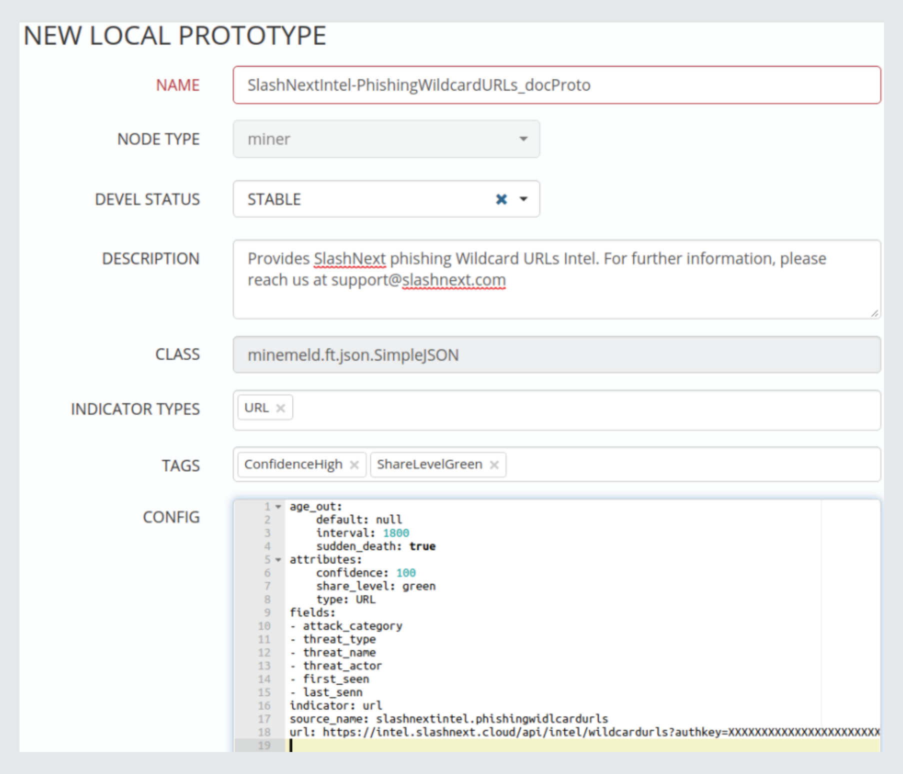
NAME	SlashNextIntel-PhishingIPs_docProto
NODE TYPE	miner
DEVEL STATUS	STABLE
DESCRIPTION	Provides SlashNext phishing IPs Intel. For further information, please reach us at support@slashnext.com
CLASS	minemeld.ft.json.SimpleJSON
INDICATOR TYPES	IPv4
TAGS	ConfidenceHigh ShareLevelGreen

CONFIG

```
age_out:
    default: null
    interval: 1800
    sudden_death: true
attributes:
    confidence: 100
    share_level: green
    type: IPv4
fields:
- attack_category
- threat_type
- threat_name
- threat_actor
- first_seen
- last_senn
indicator: hostip
source_name: slashnextintel.phishingips
url: https://intel.slashnext.cloud/api/intel/ips?authkey=XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```

https://intel.slashnext.cloud/api/intel/ips?authkey=xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx&format=json_



NEW LOCAL PROTOTYPE

NAME	SlashNextIntel-PhishingWildcardURLs_docProto
NODE TYPE	miner
DEVEL STATUS	STABLE
DESCRIPTION	Provides SlashNext phishing Wildcard URLs Intel. For further information, please reach us at support@slashnext.com
CLASS	minemeld.ft.json.SimpleJSON
INDICATOR TYPES	URL
TAGS	ConfidenceHigh ShareLevelGreen

CONFIG

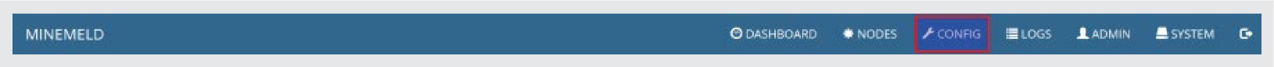
```
age_out:
    default: null
    interval: 1800
    sudden_death: true
attributes:
    confidence: 100
    share_level: green
    type: URL
fields:
- attack_category
- threat_type
- threat_name
- threat_actor
- first_seen
- last_senn
indicator: url
source_name: slashnextintel.phishingwidlcardurls
url: https://intel.slashnext.cloud/api/intel/wildcardurls?authkey=XXXXXXXXXXXXXXXXXXXXXX

```

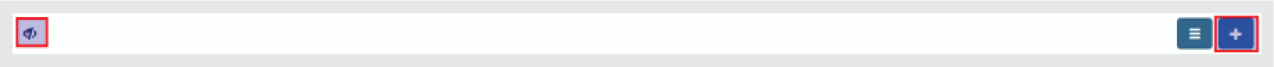
https://intel.slashnext.cloud/api/intel/wildcardurls?authkey=xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx&format=json_

2.2 |CREATING NEW SLASHNEXT MINER NODES

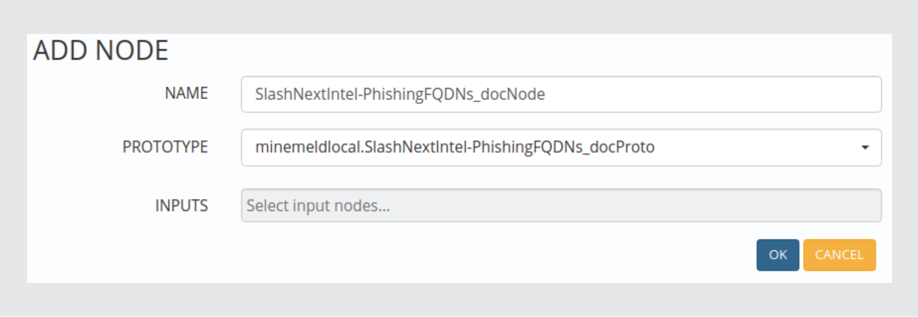
The next step is to create new SlashNext miner nodes using the newly created SlashNext miner prototypes. Click on **CONFIG** button again.



Next, click on the **eye** icon in the lower right in order to change to expert mode. Once in expert mode, a plus icon will appear allowing you to add a MineMeld node.



Click on the plus button, provide the new node with a name. From the **PROTOTYPE** drop-down, select the prototype previously created in section 2.1 as shown below



ADD NODE

NAME SlashNextIntel-PhishingFQDNs_docNode

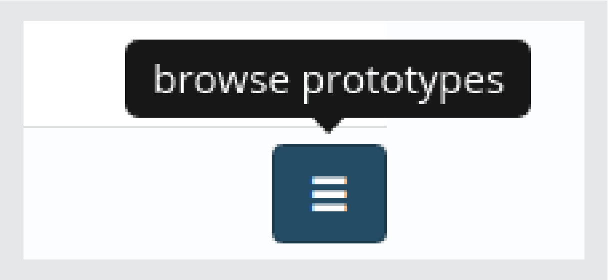
PROTOTYPE minemeldlocal.SlashNextIntel-PhishingFQDNs_docProto

INPUTS Select input nodes...

Click on OK to save the miner node and repeat the same steps to create three other miner nodes based upon rest of the miner prototypes that we created in the previous section.

2.3 |CREATING NEW SLASHNEXT AGGREGATOR PROTOTYPE

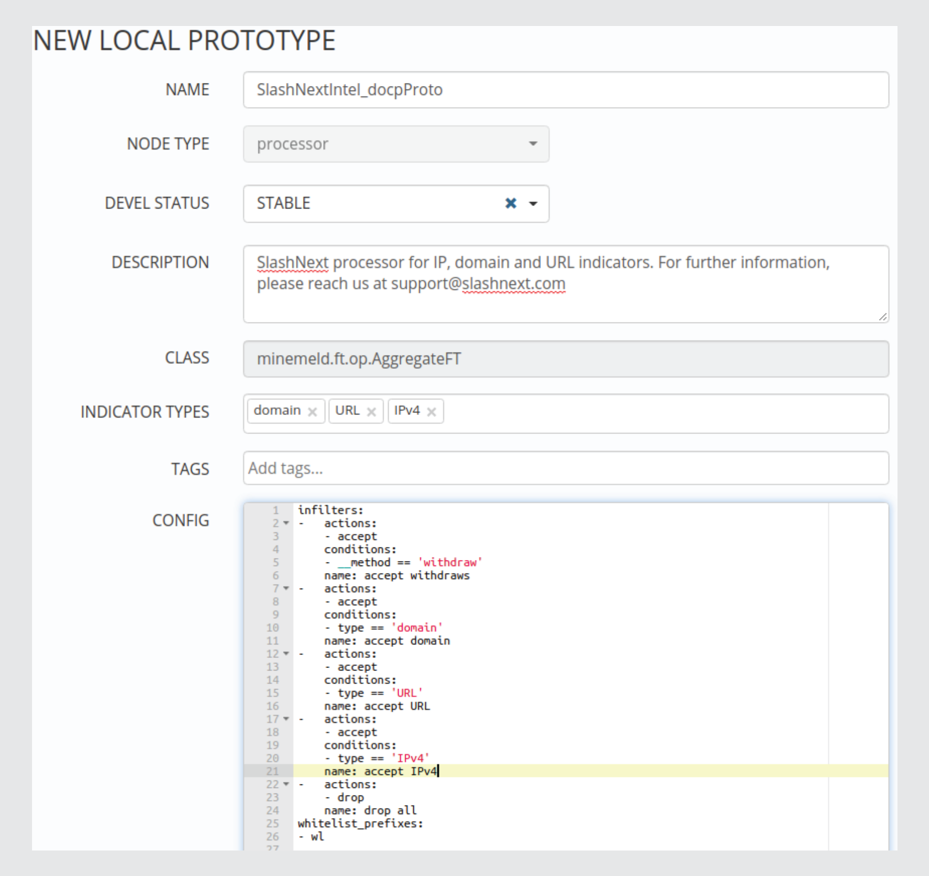
Next, a new Aggregator node (also known as a processor node) is to be created. This node will aggregate one or more miner feeds, perform de-duplication, and prepare the data to be used by an output node.
Again go to the **CONFIG** section and browse all the available prototypes by clicking the hamburger icon button.



Search for "processor" to see all the available processor prototypes. In this example, the **stdlib.aggregatorDomain** prototype is used to build the SlashNext processor prototype.

Select the **stdlib.aggregatorDomain** prototype and then click on **NEW** in the upper right-hand portion of the page to create a new aggregator node based on the one you just selected.

Modify the **NAME** to reflect the SlashNext aggregator prototype, edit the **DESCRIPTION** field and modify the **CONFIG** areas as shown below.



NEW LOCAL PROTOTYPE

NAME: SlashNextIntel_docpProto

NODE TYPE: processor

DEVEL STATUS: STABLE

DESCRIPTION: SlashNext processor for IP, domain and URL indicators. For further information, please reach us at support@slashnext.com

CLASS: minemeld.ft.op.AggregateFT

INDICATOR TYPES: domain × URL × IPv4 ×

TAGS: Add tags...

CONFIG:

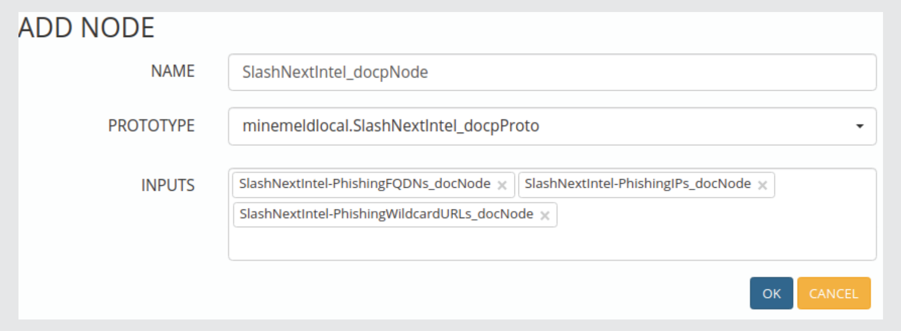
```
infilters:
-   actions:
    - accept
    conditions:
    - __method == 'withdraw'
    name: accept withdraws
-   actions:
    - accept
    conditions:
    - type == 'domain'
    name: accept domain
-   actions:
    - accept
    conditions:
    - type == 'URL'
    name: accept URL
-   actions:
    - accept
    conditions:
    - type == 'IPv4'
    name: accept IPv4
-   actions:
    - drop
    name: drop all
whitelist_prefixes:
- wl
```

2.4 |CREATING NEW SLASHNEXT AGGREGATOR NODE

Go back to **CONFIG** menu, enter expert mode as shown before, and click on the plus button to add a new aggregator node.

Give the new SlashNext aggregator node a name, and from the **PROTOTYPE** drop-down, select the prototype just created.

For the **INPUTS** field, select all four custom minor nodes previously created in section 2.2 as shown below.



ADD NODE

NAME: SlashNextIntel_docpNode

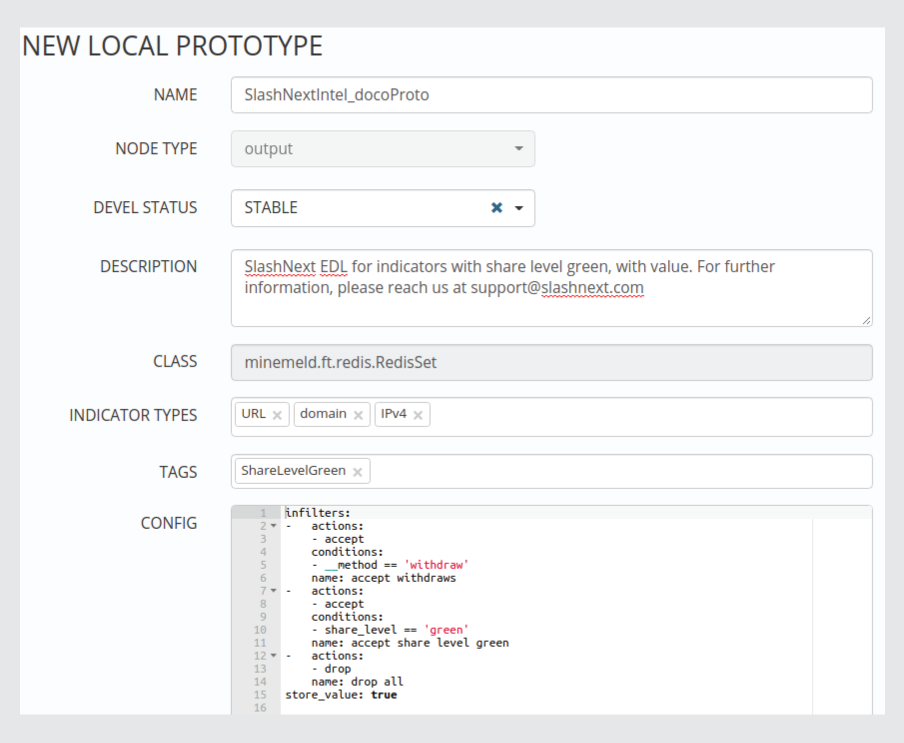
PROTOTYPE: minemeldlocal.SlashNextIntel_docpProto

INPUTS: SlashNextIntel-PhishingFQDNs_docNode × SlashNextIntel-PhishingIPs_docNode × SlashNextIntel-PhishingWildcardURLs_docNode ×

2.5 |CREATING A NEW SLASHNEXT OUTPUT PROTOTYPE

The final node to be created is the SlashNext Output node. This node will use the aggregated list data from the SlashNext aggregator node and publish it to MineMeld's internal web server so that the firewall can read the final list and use it in a policy. From the **CONFIG** menu, select the icon to browse the prototypes. In the search field, look for "output". In this example, the **stdlib.feedGreenWithValue** is used to build the SlashNext Output prototype.

Select the prototype and click **New** to create a new SlashNext output prototype based on stdlib.feedGreenWithValue
Modify the **NAME** to reflect the SlashNext output prototype, edit the **DESCRIPTION** field and modify the **CONFIG** areas as shown below.



NEW LOCAL PROTOTYPE

NAME	SlashNextIntel_docoProto
NODE TYPE	output
DEVEL STATUS	STABLE
DESCRIPTION	SlashNext EDL for indicators with share level green, with value. For further information, please reach us at support@slashnext.com
CLASS	minemeld.ft.redis.RedisSet
INDICATOR TYPES	URL × domain × IPv4 ×
TAGS	ShareLevelGreen ×
CONFIG	

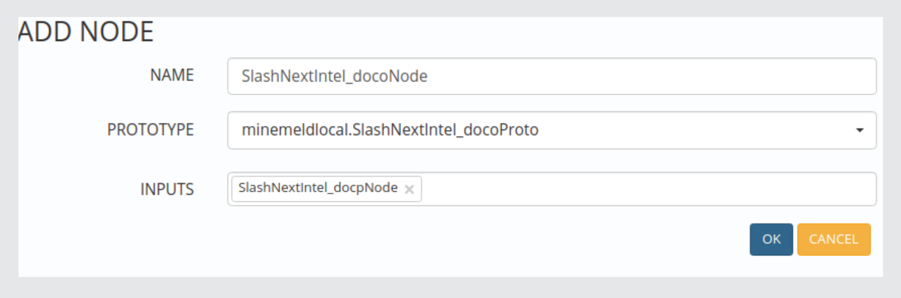
```
infilters:
-   actions:
    - accept
    conditions:
    - __method == 'withdraw'
    name: accept withdraws
-   actions:
    - accept
    conditions:
    - share_level == 'green'
    name: accept share level green
-   actions:
    - drop
    name: drop all
store_value: true
```

Finally click on **OK** to save the prototype.

2.6 |CREATING A NEW SLASHNEXT OUTPUT NODE

Go back to **CONFIG**, enter expert mode, and click on the plus button to create a new SlashNext output node based on the prototype just created. Give it a name, then select the SlashNext output prototype in the dropdown.

For the input, select the SlashNext aggregator/processor node previously created in Step 2.4 Click **OK** to save the new SlashNext output node.



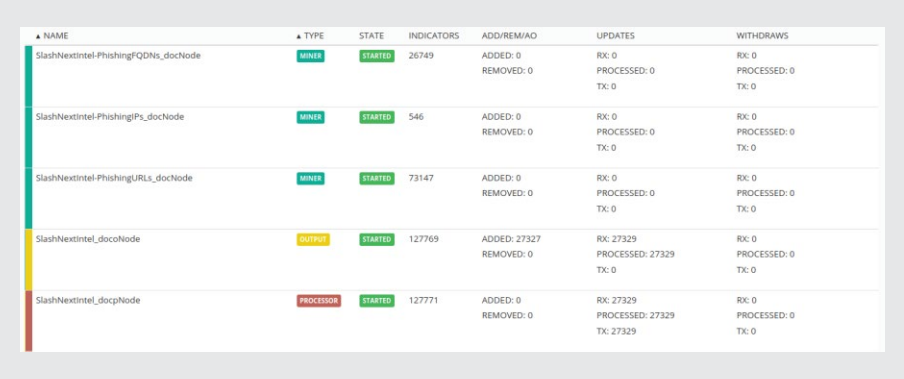
ADD NODE

NAME	SlashNextIntel_docoNode
PROTOTYPE	minemeldlocal.SlashNextIntel_docoProto
INPUTS	SlashNextIntel_docpNode ×

Return to the **CONFIG -> Prototypes**, enter "doc" in the search field which will then display all of the newly create SlashNext nodes: miners, aggregator/processor and output.

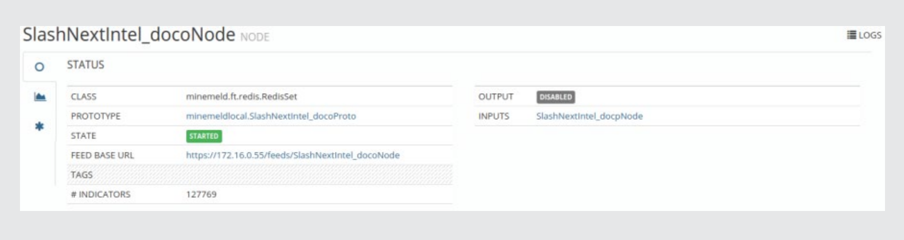
3 | FINALIZING NODE CONFIGURATIONS

After configuring all the nodes, click on the COMMIT button in the upper left-hand corner to save the node configurations and put them to work. To see if the nodes list has been created go to the Nodes menu. Click the SlashNext Output node you created.



▲ NAME	▲ TYPE	STATE	INDICATORS	ADD/REM/AO	UPDATES	WITHDRAWS
SlashNextIntel-PhishingFQDNs_docNode	MINER	STARTED	26749	ADDED: 0 REMOVED: 0	RX: 0 PROCESSED: 0 TX: 0	RX: 0 PROCESSED: 0 TX: 0
SlashNextIntel-PhishingIPs_docNode	MINER	STARTED	546	ADDED: 0 REMOVED: 0	RX: 0 PROCESSED: 0 TX: 0	RX: 0 PROCESSED: 0 TX: 0
SlashNextIntel-PhishingURLs_docNode	MINER	STARTED	73147	ADDED: 0 REMOVED: 0	RX: 0 PROCESSED: 0 TX: 0	RX: 0 PROCESSED: 0 TX: 0
SlashNextIntel_docoNode	OUTPUT	STARTED	127769	ADDED: 27327 REMOVED: 0	RX: 27329 PROCESSED: 27329 TX: 0	RX: 0 PROCESSED: 0 TX: 0
SlashNextIntel_docpNode	PROCESSOR	STARTED	127771	ADDED: 0 REMOVED: 0	RX: 27329 PROCESSED: 27329 TX: 27329	RX: 0 PROCESSED: 0 TX: 0

The **# INDICATORS** field shown in Figure 18 will begin to increment confirming proper operation.



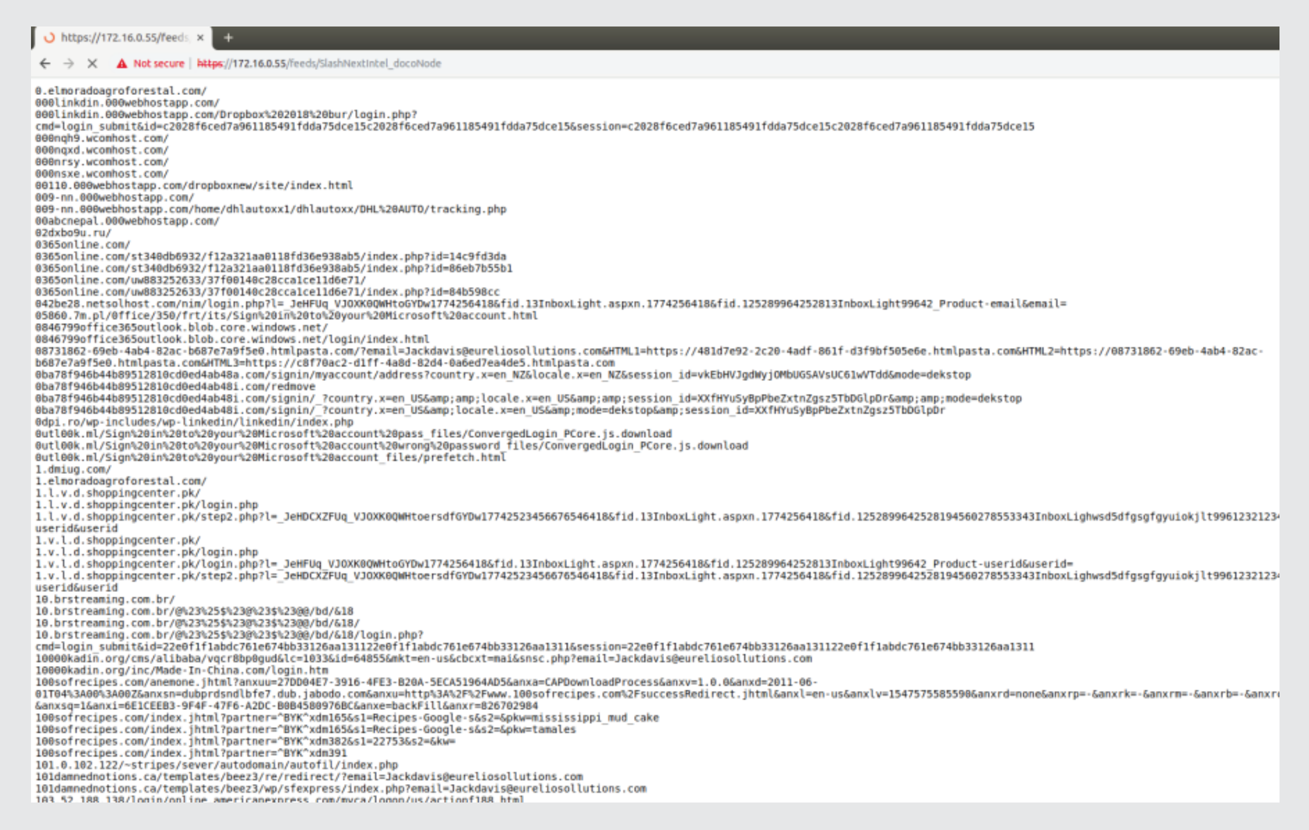
SlashNextIntel_docoNode NODE

STATUS

CLASS	minemeld.ft.redis.RedisSet
PROTOTYPE	minemeldlocal.SlashNextIntel_docoProto
STATE	STARTED
FEED BASE URL	https://172.16.0.55/feeds/SlashNextIntel_docoNode
TAGS	
# INDICATORS	127769

OUTPUT	DISABLED
INPUTS	SlashNextIntel_docpNode

The **# INDICATORS** field shown in Figure 18 will begin to increment confirming proper operation.



https://172.16.0.55/feeds ×
Not secure | https://172.16.0.55/feeds/SlashNextIntel_docoNode

```
0.elmoradoagroforestal.com/
000linkdin.000webhostapp.com/
000linkdin.000webhostapp.com/Dropbox%202018%20bur/login.php?
cmd=login_submit&id=c2028f6ced7a961185491fdda75dce15c2028f6ced7a961185491fdda75dce15&session=c2028f6ced7a961185491fdda75dce15c2028f6ced7a961185491fdda75dce15
000nqh9.wcomhost.com/
000nqxd.wcomhost.com/
000nrsy.wcomhost.com/
000nsxe.wcomhost.com/
00110.000webhostapp.com/dropboxnew/site/index.html
009-nn.000webhostapp.com/
009-nn.000webhostapp.com/home/dhlautoxx1/dhlautoxx/DHL%20AUTO/tracking.php
00abcnepal.000webhostapp.com/
02dxbo9u.ru/
0365online.com/
0365online.com/st340db6932/f12a321aa0118fd36e938ab5/index.php?id=14c9fd3da
0365online.com/st340db6932/f12a321aa0118fd36e938ab5/index.php?id=86eb7b55b1
0365online.com/uw883252633/37f00140c28cca1ce11d6e71/
0365online.com/uw883252633/37f00140c28cca1ce11d6e71/index.php?id=84b598cc
042be28.netsolhost.com/nim/login.php?l=_JeHFUq_VJOXK0QWHtoGYDw1774256418&fid.13InboxLight.aspxn.1774256418&fid.125289964252813InboxLight99642_Product-email&email=
05860.7m.pl/0ffice/350/frt/its/Sign%20in%20to%20your%20Microsoft%20account.html
0846799office365outlook.blob.core.windows.net/
0846799office365outlook.blob.core.windows.net/login/index.html
08731862-69eb-4ab4-82ac-b687e7a9f5e0.htmlpasta.com/?email=Jackdavis@eureliosollutions.com&HTML1=https://481d7e92-2c20-4adf-861f-d3f9bf505e6e.htmlpasta.com&HTML2=https://08731862-69eb-4ab4-82ac-
b687e7a9f5e0.htmlpasta.com&HTML3=https://c8f70ac2-d1ff-4a8d-82d4-0a6ed7ea4de5.htmlpasta.com
0ba78f946b44b89512810cd0ed4ab48a.com/signin/myaccount/address?country.x=en_NZ&locale.x=en_NZ&session_id=vkEbHVJgdWyjOMbUGSAVsUC61wVTdd&mode=dekstop
0ba78f946b44b89512810cd0ed4ab48i.com/redmove
0ba78f946b44b89512810cd0ed4ab48i.com/signin/_?country.x=en_US&amp;amp;locale.x=en_US&amp;amp;session_id=XXfHYuSyBpPbeZxtnZgsz5TbDGlpDr&amp;amp;mode=dekstop
0ba78f946b44b89512810cd0ed4ab48i.com/signin/_?country.x=en_US&amp;locale.x=en_US&amp;mode=dekstop&amp;session_id=XXfHYuSyBpPbeZxtnZgsz5TbDGlpDr
0dpi.ro/wp-includes/wp-linkedin/linkedin/index.php
0utl00k.ml/Sign%20in%20to%20your%20Microsoft%20account%20pass_files/ConvergedLogin_PCore.js.download
0utl00k.ml/Sign%20in%20to%20your%20Microsoft%20account%20wrong%20password_files/ConvergedLogin_PCore.js.download
0utl00k.ml/Sign%20in%20to%20your%20Microsoft%20account_files/prefetch.html
1.dmiug.com/
1.elmoradoagroforestal.com/
1.l.v.d.shoppingcenter.pk/
1.l.v.d.shoppingcenter.pk/login.php
1.l.v.d.shoppingcenter.pk/step2.php?l=_JeHDCXZFUq_VJOXK0QWHtoersdfGYDw17742523456676546418&fid.13InboxLight.aspxn.1774256418&fid.1252899642528194560278553343InboxLighwsd5dfgsgfgyuiokjlt9961232123
userid&userid
1.v.l.d.shoppingcenter.pk/
1.v.l.d.shoppingcenter.pk/login.php
1.v.l.d.shoppingcenter.pk/login.php?l=_JeHFUq_VJOXK0QWHtoGYDw1774256418&fid.13InboxLight.aspxn.1774256418&fid.125289964252813InboxLight99642_Product-userid&userid=
1.v.l.d.shoppingcenter.pk/step2.php?l=_JeHDCXZFUq_VJOXK0QWHtoersdfGYDw17742523456676546418&fid.13InboxLight.aspxn.1774256418&fid.1252899642528194560278553343InboxLighwsd5dfgsgfgyuiokjlt9961232123
userid&userid
10.brstreaming.com.br/
10.brstreaming.com.br/@%23%25$%23@%23$%23@@/bd/&18
10.brstreaming.com.br/@%23%25$%23@%23$%23@@/bd/&18/
10.brstreaming.com.br/@%23%25$%23@%23$%23@@/bd/&18/login.php?
cmd=login_submit&id=22e0f1f1abdc761e674bb33126aa131122e0f1f1abdc761e674bb33126aa1311&session=22e0f1f1abdc761e674bb33126aa131122e0f1f1abdc761e674bb33126aa1311
10000kadin.org/cms/alibaba/vqcr8bp0gud&lc=1033&id=64855&mkt=en-us&cbcxt=mai&snsc.php?email=Jackdavis@eureliosollutions.com
10000kadin.org/inc/Made-In-China.com/login.htm
100sofrecipes.com/anemone.jhtml?anxuu=27DD04E7-3916-4FE3-B20A-5ECA51964AD5&anxa=CAPDownloadProcess&anxv=1.0.0&anxd=2011-06-
01T04%3A00%3A00Z&anxsn=dubprdsndlbfe7.dub.jabodo.com&anxu=http%3A%2F%2Fwww.100sofrecipes.com%2FsuccessRedirect.jhtml&anxl=en-us&anxlv=1547575585598&anxrd=none&anxrp=-&anxrk=-&anxrm=-&anxrb=-&anxr
&anxsq=1&anxi=6E1CEEB3-9F4F-47F6-A2DC-B0B4580976BC&anxe=backFill&anxr=826702984
100sofrecipes.com/index.jhtml?partner=^BYK^xdm165&s1=Recipes-Google-s&s2=&pkw=mississippi_mud_cake
100sofrecipes.com/index.jhtml?partner=^BYK^xdm165&s1=Recipes-Google-s&s2=&pkw=tamales
100sofrecipes.com/index.jhtml?partner=^BYK^xdm382&s1=22753&s2=&kw=
100sofrecipes.com/index.jhtml?partner=^BYK^xdm391
101.0.102.122/~stripes/sever/autodomain/autofil/index.php
101damnednotions.ca/templates/beez3/re/redirect/?email=Jackdavis@eureliosollutions.com
101damnednotions.ca/templates/beez3/wp/sfexpress/index.php?email=Jackdavis@eureliosollutions.com
103.52.188.138/login/online.americanexpress.com/myca/logon/us/actionf188.html
```