

NSPEPI tool for policy conversion prior to Citrix ADC 12.1-51.16 release

You can convert a Classic expression to the Advanced policy expression by using the NSPEPI conversion tool. You can also use the tool to convert all the classic expressions in the Citrix ADC configuration to the Advanced policy (with the exception of Citrix ADC entities that currently support only classic expressions).

The conversion tool does not convert policies configured for the following features because the features currently support only classic policies:

- Authentication, Pre-authentication
- SSL
- Cache redirection
- VPN (session, traffic, and tunnel traffic)
- Content filtering (The responder feature not only provides you with functionality that is equivalent to that provided by the content filtering feature but also surpasses the content filtering feature in the use cases that it supports. Additionally, responder supports the more powerful default syntax for policy expressions.)

The following Citrix ADC features support both classic and Advanced policy expression and, therefore, support the conversion of classic expressions to Advanced policy expressions:

- Application firewall policies
- Authorization policies
- Named expressions
- Compression policies
- Content switching policies
- User-defined, rule-based tokens/persistency (the `-rule` parameter value that is specified for a load balancing virtual server)

About the conversion process

When parsing a Citrix ADC configuration file, the conversion tool performs the following actions:

1. In commands that create classic named expressions, the conversion tool replaces the names of the classic expressions with Advanced policy expression.
2. In commands that support only the classic policy, if classic named expressions are used, the conversion tool replaces the names of the classic expressions with the actual classic expressions they represent. This action ensures that the names of expressions in classic-only features do not reference the Advanced policy expression created from Step 1.
3. In commands associated with entities that support both the classic syntax and the Advanced policies, the conversion tool replaces all classic expressions in commands with Advanced policy expressions.

Example:

Consider the following sample configuration commands:

```
COPY
```

```
add policy expression ne_c1 "METHOD == GET"
add policy expression ne_c2 "ne_c1 || URL == /*.htm "
add filter policy pol1 -rule "ne_c2" -reqAction YES
add cmp policy pol2 -rule "REQ.HTTP.HEADER Accept CONTAINS
'text/html'" -resAction COMPRESS
add cmp policy pol3 -rule "ne_c1 || ne_c2" -resAction GZIP
```

In the commands that create the classic named expressions ne_c1 and ne_c2, the tool replaces the names of the expressions with actual Advanced policy expression. This action, which corresponds to Step 1 described earlier, results in the following commands:

```
add policy expression ne_c2 "HTTP.REQ.METHOD.EQ(\"GET\") "
add policy expression ne_c1 "HTTP.REQ.URL.SUFFIX.EQ(\"htm\") "
```

The filter policy command supports only the classic syntax. Therefore, the conversion tool replaces the classic named expression ne_c1 with the actual classic expression it represents. Note that the tool replaces ne_c1 in the expression for ne_c2, and then replaces ne_c2 in the filter policy with the classic expression. This action, which corresponds to Step 2 described earlier, results in the following command:

```
add filter policy pol1 -rule "METHOD == GET || URL == /*.htm" -
reqAction YES
```

The compression feature supports both classic and Advanced policy expression. Therefore, in the command that creates the compression policy pol2, the conversion tool replaces the expression with Advanced policy expression. This action, which corresponds to Step 3 described earlier, results in the following command:

```
add cmp policy pol2 -rule
"HTTP.REQ.HEADER(\"Accept\").AFTER_STR(\"text/html\").LENGTH.GT(0) " -
resAction COMPRESS
```

The command that creates the compression policy pol3 is unaffected by the conversion process because, after the conversion process is complete, ne_c1 and ne_c2 reference the Advanced policy expression that results from Step 1.

Client security messages are not supported in the newer default policy format and, therefore, are lost. The SYS.EVAL_CLASSIC_EXPR function is replaced with a Advanced policy expression. The following entities support the SYS.EVAL_CLASSIC_EXPR function:

- DNS policies
- Rate limit selectors
- Cache selectors
- Cache policies
- Content switching policies
- Rewrite policies
- URL transformation policies
- Responder policies
- Application firewall policies
- Authorization policies
- Compression policies
- CVPN access policies

After performing the conversion, the tool saves the changes in a new configuration file. The new configuration file is created in the directory in which the input file exists. The name of the new configuration file is the same as the name of the input configuration file except for the string new_ used as a prefix. Conversion warnings are reported in a warning line at the end of the screen output. Additionally, a warning file is created in the directory in which the input configuration file resides.

Convert expressions

You can use the nspepi tool to convert a single classic expression to the Advanced policy expression. The nspepi tool must be run from the shell prompt on the Citrix ADC appliance.

Convert a classic expression to the Advanced policy by using the CLI

At the shell prompt, type:

```
nspepi -e "<classic expression>"
```

Example:

```
root@NS# nspepi -e "REQ.HTTP.URL == /*.htm"  
  
"HTTP.REQ.URL.REGEX_MATCH(re#/(.*)\.htm#)"
```

Convert a Citrix ADC configuration file

You can use the nspepi tool to convert all the classic expressions in a Citrix ADC configuration file to the Advanced policy (except for those commands that do not support the Advanced policy). The nspepi tool must be run from the shell prompt on the Citrix ADC appliance.

Convert all the classic expressions in a Citrix ADC configuration file to the default syntax by using the CLI

At the shell prompt, type:

```
nspepi -f "<ns config file>" -v
```

Example

```
root@NS# nspepi -f ns.conf
OUTPUT: New configuration file created: new_ns.conf
OUTPUT: New warning file created: warn_ns.conf
WARNINGS: Total number of warnings due to bind commands: 18
WARNINGS: Line numbers which has bind command issues: 305, 306, 706, 707,
708, 709, 710, 711, 712, 713,
714, 715, 767, 768, 774, 775, 776, 777
root@NS#
```

Conversion warnings

When classic expressions that are included in CLI commands are upgraded to the Advanced policy, the number of characters in the expression might exceed the 1499-character limit. The commands that include expressions longer than 1499 characters fail when the configuration is being applied. You must manually update these commands.

In addition, multiple classic policies can be bound to a given bind point with priority 0 or with equal priority, but the Advanced policy infrastructure does not support a priority value of 0 or policies with the same priority at a given bind point. These commands fail when the configuration is being applied. The commands must be updated manually with the correct priority values.

The line numbers of lines that threw a warning during conversion are listed at the end of the output in a warning line. In addition, a warning file is created in the same directory as the one in which the old and new configuration files reside. The name of the warning file is the same as the name of the input configuration file except that the string `warn_` is added as a prefix.