



Geek Speak Live!

Unified Gateway in-a-box

GitHub **Quick Reference** 2017/v1.0.0

Thorsten Rood, CTP









thorsten@rood.cc

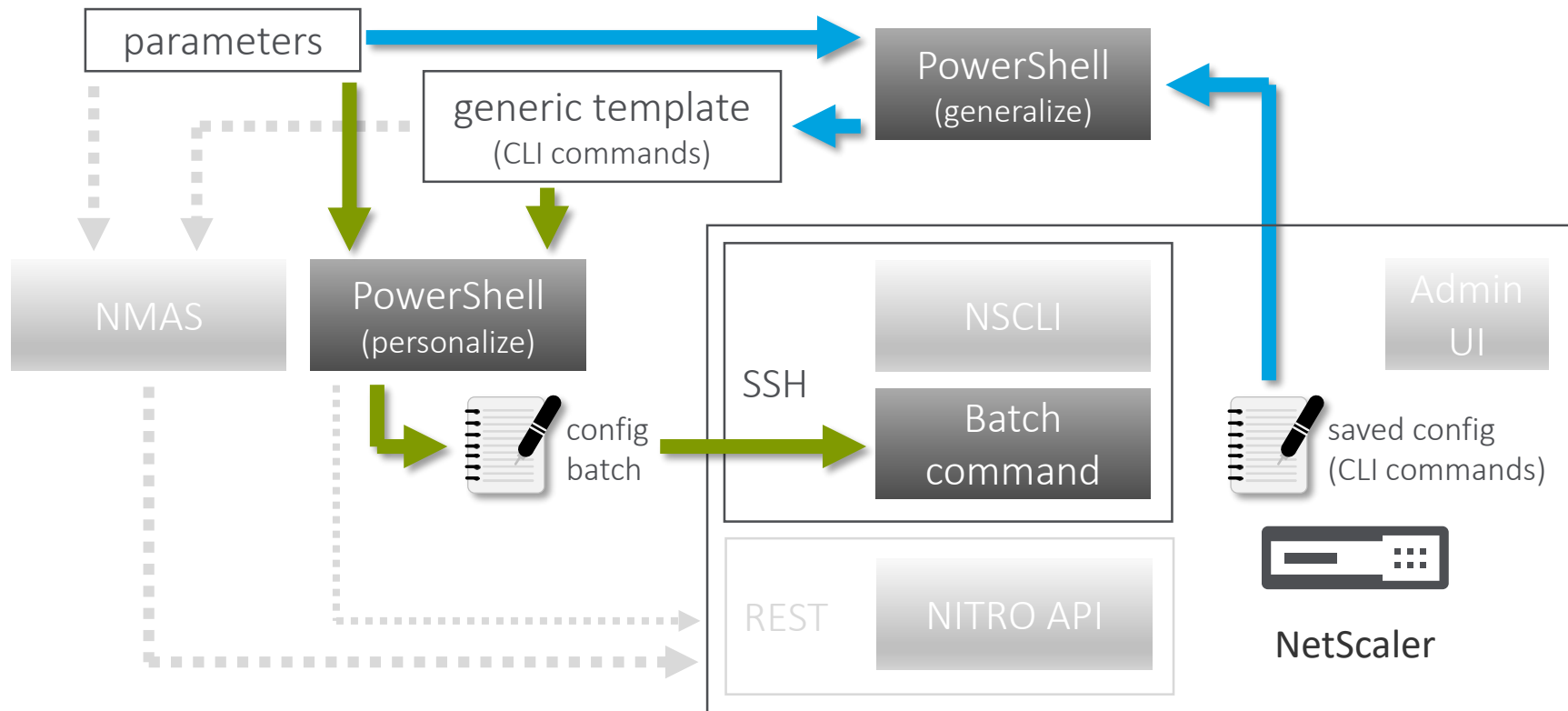


@thorstenrood



Introducing “UG-in-a-box”, release v1.0

 Admin
  available now
 Engineer
   prototype phase
 long-term option



UG-in-a-box

Introducing community edition

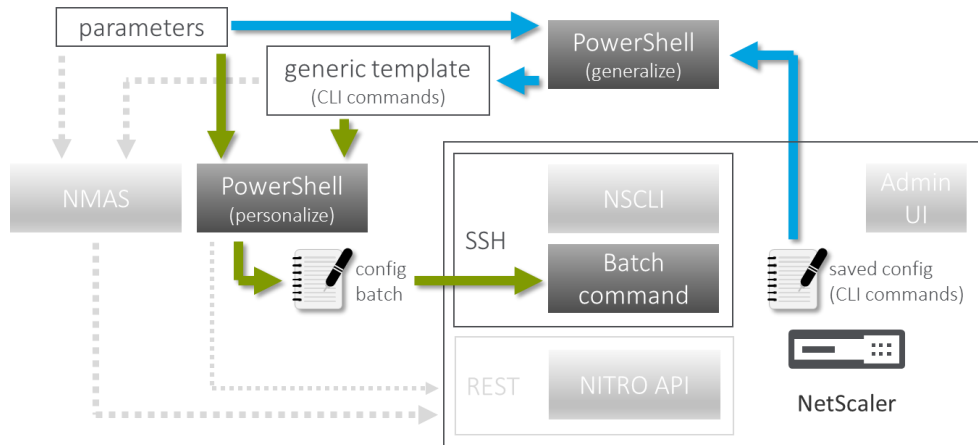
- Unified Gateway in a box (UG-in-a-box) community edition available now!

- Released 04/2017, version 1.0
- Free download, free for commercial use

- It's a verification release (covers today's session)!

- Does it work for you (usage at own risk)?
- Do you need more specific guidance/training?
- Do you want more modules (puzzle tiles a.k.a. use-cases)?
- Anybody joining the technical QA team?

- Let's get into the details...



<https://www.rood.cc/UGiab/>
(redirects to GitHub)

CITRIX®
Geek Speak Live!



UG in a box

Execution reference (raw)



UG in a box

Command line reference



- **Step 1 – Customization/Preparation:** `parameter_file(variables).csv`
 - `<###param1###/>,yourvalue1`
 - `<###param2###/>,yourvalue2`
 - The `<###...###/>` will be replaced with “valueX” in forward processing mode, and opposite in reverse mode.
 - Especially on reverse mode, make sure the specific configuration doesn’t overlap (not even subsets).
 - CSV content is handled case-sensitive.

UG in a box

Command line reference



- **Step 2 – Apply customization parameters into template files:** compilesettings.ps1

- This is the core “for-each” replace tool that combines the available generic modules with parameters and creates a specific configuration batch
- **placeholders:** (mandatory) the variables file as discussed before
- **modulepath:** the directory where all .conf modules are located, files will be concatenated alphabetically. Operates on current directory, if omitted
- **outputfile:** the concatenated result of all .conf modules in a single flat file
- **detailed:** defaults to false. If set to true, each .conf module will be additionally written as a .patched file
- You need to specify either **outputfile** or **detailed=\$true**, or both
- **reverse:** defaults to false. If set to true, the placeholders are executed in mirrored replacement sequence (right-to-left). This is an engineering option to generalize a given specific .conf into template(s).
- All output files will respect UNIX end of line encoding (LF)

UG in a box

Command line reference



- **Step 3 – Deploy files and execute batch:** `fulldeploy.cmd`

- This is a helper to load (binary) files on the appliance, such as UG logos, certificates and the batch and then executes the UG-in-a-box configuration on a “factory” instance NetScaler
- Requires a locally installed PuTTY package with `pscp.exe` and `plink.exe` in default folder `%programfiles(x86)%\putty\`
- Needs to be called with five mandatory parameters in exact sequence as outlined below:
- `nsip`: appliance mgmt IP address
- `nsrootPassword`: guess what...
- `PNGpath`: absolute or relative path to the repository where UG icons (.png) are stored
- `CERTpath`: absolute or relative path to the repository where certificate files (.crt, .key) are stored
- `batch`: outputfile from preceding PowerShell customization script
- Like this: `fulldeploy.cmd 10.64.27.140 Passw0rd logos*.png certs*. * mybatch.conf`

UG in a box

Known issues and limitations



- DHparam key
 - generation API timeouts often for 2K key, especially when used inside batch command (even on CLI)
 - Maybe prestage the keyfile, depending on CNS version
 - *[incident filed, unsolved so far]*