

TeamsUpdated

Description

This software checks all of Microsoft Teams' config URLs (e.g. "https://config.teams.microsoft.com/config/v1/MicrosoftTeams/") and checks for Microsoft's static URLs (e.g. "https://staticsint.teams.cdn.office.net/production-windows-x64/1.8.00.18552/Teams_windows_x64.exe") various releases of Microsoft Teams. Upon finding a new release, it will begin the installation and uninstallation process for that release and stores the URL in a data.json file afterward to prevent reinstalling the same version in the future. This software features strong error handling and will indicate when manual intervention is needed via a "MANUAL INTERVENTION NEEDED.txt" file in the data path. Be sure to check that path frequently.

Example usages

Runs in CMD or powershell. You could also just double click the EXE. The program will run repeatedly unless interrupted with the **CTRL + C** interrupt signal.

```
TeamsUpdated.exe -datapath C:\Users\neil\Desktop\TeamsUpdated -consoleloglevel verbose  
-fileloglevel information
```

```
TeamsUpdated.exe -disablefilelog -disableconsolelog -interval 60
```

```
TeamsUpdated.exe -platform win_arm64 -avoidenv dev,gcc_high,dod -avoidrings  
ring0,ring0_s
```

```
TeamsUpdated.exe -noloop -onlyrings ring0,ring0_s,ring1 -onlyenvs dev,prod,life
```

```
TeamsUpdated.exe -isclassic -onlyrings ring3_6 -onlyenvs prod
```

```
TeamsUpdated.exe -onlyrings ring3 -onlyenvs prod -objectid 3a7cf1d3-06fa-4ead-bf45-  
a6286ff2620a -tenantid 72f988bf-86f1-41af-91ab-2d7cd011db47
```

```
TeamsUpdated.exe -isexperimental
```

```
TeamsUpdated.exe
```

Arguments

Capitalization does not matter.

name	alias	description	default	values
<code>-interval</code>	<code>-minuteInterval</code>	The interval in minutes before doing running again.	30.0	Any decimal value (non-negative)
<code>-datapath</code>		The path where all data files are placed. Dat files include data.json, "MANUAL INTERVENTION NEEDED.txt", and log.txt	<code>C:\TeamsUpdatedData</code>	Any existng directory path. Directory must already exist or an error will be thrown.
<code>-consoleLogLevel</code>		Level of detailed logs the exe will output to the console.	Information	Verbose, Debug, Information, Warning, Error, Fatal
<code>-fileLogLevel</code>		Level of detailed logs the exe will output to the log file.	Information	Verbose, Debug, Information, Warning, Error, Fatal
<code>-noFileLog</code>	<code>-disableFileLog</code>	Disables file logging if present.	False	None (True if flag is present)
<code>-noConsoleLog</code>	<code>-disableConsoleLog</code>	Disables console logging if present.	False	None (True if flag is present)
<code>-noLoop</code>	<code>-disableLoop</code>	Disables looping of checks. Recommended for using scheduler.	False	None (True if flag is present)
<code>-isClassic</code>		Will only check V1 in the 1415 config url platform id.	False	None (True if flag is present)

name	alias	description	default	values
<code>-isexperimental</code>		Will only check for experimental releases of v1. This requires tootally different logic; details below.	<code>False</code>	None (<code>True</code> if flag is present)
<code>-platform</code>		The platform to install updates for.	<code>Win_x64_Win_x86</code>	<code>Win_x64_Win_x86, Win_x64, Win_x86, Win_arm64, OSX_x64_OSX_arm64</code>
<code>-avoidrings</code>		Update rings to avoid checking.		<code>Ring0, Ring0_S, Ring1, Ring1_5, Ring1_6, Ring2, Ring3, Ring3_6, Ring3_9, General, General_Gcc</code>
<code>-onlyrings</code>		Will only check the listed rings.	<code>All of them</code>	<code>Ring0, Ring0_S, Ring1, Ring1_5, Ring1_6, Ring2, Ring3, Ring3_6, Ring3_9, General, General_Gcc</code>
<code>-avoidenvironments</code>	<code>-avoidenv, -avoidenvs</code>	Release environments to avoid checking.	<code>dod, gcchigh</code>	<code>Dev, Prod, Life, Gcc, Gcchigh, Dod, Gallatin</code>
<code>-onlyenvironments</code>	<code>-onlyenv, -onlyenvs</code>	Will only check the set environments.	<code>All of them</code>	<code>Dev, Prod, Life, Gcc, Gcchigh, Dod, Gallatin</code>
<code>-tenantid</code>		Sets the tenant id URL serach parameter to be used with the MS config API request.	<code>None</code>	Any string, though the API is expecting a GUID
<code>-objectid</code>		Sets the object id URL serach parameter to be used with the MS config API request.	<code>None</code>	Any string, though the API is expecting a GUID

Data Files

All data files are stored in the directory specified by the `-datapath` argument (default: `C:\TeamsUpdatedData`). The following files are used:

- `data.json`: Tracks which Teams versions have already been installed and uninstalled to prevent duplicate processing.
- `MANUAL INTERVENTION NEEDED.txt`: Created when a manual step is required. The program will not proceed until this file is deleted.
- `log.txt`: Contains logs of all operations, errors, and informational messages.
- `<platform>.txt`: Used in experimental mode to track the last checked build number for each platform.

Be sure to check the data path regularly for any manual intervention messages.

-IsExperimental for Teams Version 1

New releases of Teams classic (aka 1.0) experimental are not provided by Microsoft in the config API. Instead, this software will check for new releases by hitting all the possible future static URLs (e.g. "https://staticsint.teams.cdn.office.net/production-windows-x64/1.8.00.18552/Teams_windows_x64.exe") and checking the HTTP response code. This requires more time to be performed as the script will check 384 new versions at a time. Whenever a successful response is received, the software will save the build number to the data path in a file named `<platform>.txt` (e.g. `win_x64.txt`). This is referenced in the next run to prevent duplicate checks. If a new build is found, it will be installed and uninstalled like normal.