

Working an RMF package at the scale it actually is

How STIGrevolution handles hundreds of STIG checklists at once, finds the problems in them before a reviewer does, and keeps the eMASS POA&M and Assessment Procedure results in step with the testing behind them.

STIG Solution LLC · stigsolution.com/stigrevolution/

AT A GLANCE

What it is	A Windows application for working STIG checklists in bulk, with analysis across the whole set and eMASS POA&M and Assessment Procedure (AP) results built from your testing
Works with	CKL and CKLB checklists, SCAP results, ACAS (.nessus) files, the eMASS POA&M export, the eMASS TR Export, and policy results files you create
Produces	Checklists, Excel reports, POA&M updates, and the eMASS TR Import
Runs on	Windows 10 and 11; Windows Server 2012 R2, 2016 and 2019; .NET Framework 4.7.1 or higher. Single-file EXE or MSIX.
Your data	Stays on your machine. The only outbound connection is the license check over port 443.
Licensing	14-day free trial with no license key needed to start, then 365 days from date of purchase

Summary

An RMF package is not a pile of checklists. It is a set of artifacts that have to agree with each other: the hardware and software list, the test plan, the checklists, the scan results, and the eMASS POA&M built from all of them. A single Windows server carries six to twelve checklists. A package runs from fifty or a hundred into the low hundreds.

At that volume the standard tools, which open one checklist at a time, stop being slow and start being the reason things go wrong. Teams fill the gap with spreadsheets, PowerShell, and Python, which hold together until the package is large, the deadline is close, or the person who wrote the scripts has moved on.

STIGrevolution works the whole set at once. It creates, updates, and edits checklists in bulk; flags missing host information, unreviewed items, and closed items with no finding details as soon as the files are loaded; analyzes every checklist together; and writes the eMASS POA&M updates and Assessment Procedure results from the testing you have loaded. This paper describes how.

The problem: hundreds of files that have to agree

The work that sends a package back for rework is rarely a single wrong answer. It is disagreement between artifacts: a checklist built on a superseded STIG, a host missing from the test plan, an item closed with nothing in its finding details, a POA&M item that no longer matches the result behind it.

None of that is visible one file at a time, and the standard tools work one file at a time:

- Checklists are created from SCAP results one at a time.
- “Not Reviewed” items have to be hunted across hundreds of files, with no way to see them together.
- Host information is corrected one checklist at a time.
- When a new STIG version is released, the way to find out which checklists went stale is to open them.

So teams build their own way through. Spreadsheets, PowerShell, and Python all work, up to a point, and each brings its own failure:

- **They don’t scale.** What works for forty checklists breaks at four hundred.
- **They need a particular skill.** The process runs only as long as someone on the team can write and maintain the scripts. That is a staffing constraint and a key-person risk.
- **They can’t be turned into a process.** Nothing repeatable, nothing that can be documented and handed to a new team member.

The cost is not mainly hours. It is rework: a package returned with mistakes, an RMF timeline that stretches, and an ATO date that slips because the artifacts drifted apart and nothing in the toolchain noticed.

STIGrevolution started the same way, as tooling written by people doing this work for the packages in front of them. The difference is that it was built to be handed to anyone on the team.

Bulk operations across the whole set

In STIGrevolution the unit of work is the set of checklists, not the file.

Create checklists from SCAP output in bulk

Load the SCAP results for every host and create every checklist in one operation. Fifty SCAP results become fifty checklists in one step, not fifty.

Move to new STIG versions in bulk

When a quarter's STIG revisions are released, update every affected checklist at once instead of discovering them one file at a time. STIGrevolution ships with the current quarter's STIG revisions file, the same file STIG Solution publishes free each quarter.

Edit across assets

Set status, finding details, and comments for the same item on many assets in a single pass, rather than repeating the same edit in every file.

Duplicate against hostname and IP lists

Stand up checklists for a whole set of systems from a hostname and IP list you already have.

Load files the way you have them

Checklists (CKL and CKLB), ACAS scans (.nessus), and SCAP results each load from individual files, folders, or ZIP archives through one Open button per page. STIGrevolution remembers the folders you load from, so the same set can be reloaded in one click next time.

Measured time savings. Updating 100 checklists to a new STIG version, or creating 100 checklists from 100 SCAP results, takes under a minute in STIGrevolution, against two to three hours one file at a time.

Find the problems before a reviewer does

Load your files and STIGrevolution flags the problems straight away: missing host information, items still Not Reviewed, and items closed with no finding details. Every issue is visible up front, across the whole set, and can be corrected before anything is built on top of it.

Review interface

Check and correct host information, surface every “Not Reviewed” item, and find items closed with no finding details, across the whole set rather than file by file.

Analysis

Sort, filter, and group across every loaded checklist at once, then export the view as a report rather than as raw data that still has to be shaped in Excel.

Test plan coverage

SCAP results are matched against the entries in your test plan. Results whose hostnames are fully qualified domain names match their test plan entries too, and are marked as FQDN matches so they can be told apart from exact ones. The test plan is an inventory of every checklist and SCAP result the system should have, so a host that was planned but not tested, or tested but not planned, stands out.

The eMASS POA&M, kept in step with the results

A POA&M that has drifted from the results behind it is a specific, recognizable, and expensive failure: items that should have closed are still open, findings that should be on it are missing, and nobody finds out until a reviewer does.

STIGrevolution imports the eMASS POA&M export alongside your checklists and ACAS files, shows where they disagree, and generates the updates.

- **The 2026 eMASS POA&M template** is supported for import and export, including the Summary, Date Identified, and Actual Cost columns and the renamed Vulnerability Description and Resources columns. POA&M files in the older layout still import; both generations are recognized automatically.
- **Exports** follow the template, with Navy, Army, Air Force, and Army-with-financial-columns variants. The STIG rule title goes into Summary and the plain rule discussion into Vulnerability Description.
- **Army and Air Force POA&Ms** can be viewed and searched alongside the rest.

Assessment Procedure results, written from your testing

eMASS holds a result for every Assessment Procedure (AP) and exports them in the TR Export, an Excel spreadsheet you update and upload to change the results in eMASS. Filling it in by hand means pasting standard wording and test evidence into hundreds of APs, one at a time.

STIGrevolution writes those results from the testing you already have loaded.

1. **Every result that answers the AP.** Any loaded result tied to a CCI, AP, or control counts, compliant or not, from STIG checklists and ACAS alike. Policy results for each AP are kept in a file you create and edit locally, like a checklist, and aggregate with the rest.
2. **In your order, in your words.** Choose the order results are listed in and add text before and after them: the standard wording you would otherwise paste into each AP by hand.
3. **Written to fit the field.** Technical results are aggregated per check (ID, STIG, title, and a host count such as “12 of 14 hosts compliant”). Each result is capped at 4,000 characters, the size of the eMASS field, and edited down to fit when the full listing won’t.

Status follows the evidence. When results from several sources are combined for one AP, Non-Compliant takes precedence over Compliant, and an AP is reported as Not Applicable only when all of its results are. APs that are Not Applicable on every host can be added from your loaded checklists in one step.

When eMASS and the POA&M disagree

Load the TR Export beside the POA&M and STIGrevolution shows every place they are out of sync, including Not Applicable APs that still carry an Ongoing POA&M item. It then generates a TR Import: the updated spreadsheet that brings eMASS back in line.

Where STIGrevolution fits

STIG Solution makes three applications for the same person: the one accountable for the package.

	STIGreviewer	STIGrevolution	NESSviewer
Job	Edit and validate checklists, one at a time	Work checklists in bulk and run the package around them	Turn Nessus scan data into package artifacts
Price	Free, permanently	Licensed	Licensed

STIGreviewer, the free replacement for DISA STIG Viewer, edits checklists one at a time. STIGrevolution does that work in bulk, across every checklist at once, and adds the analysis, POA&M, and Assessment Procedure layer on top.

What’s next. The next release of STIGrevolution moves to .NET MAUI, adds macOS, and brings the baseline module already in the NESSviewer beta: the hardware and software list and the test plan, shared between the two applications. It enters beta after NESSviewer’s.

Deployment, data handling, and licensing

STIGrevolution is built for environments where software is scrutinized before it is allowed to run.

- **Two packages, one application.** A single-file EXE that runs without installing, or an MSIX for managed deployment through the tools your organization already uses.
- **Your data stays on your machine.** Checklists, scans, and POA&Ms are read and written locally.
- **One outbound connection.** The license check over port 443, to licensing.stigsolution.com, which runs in Microsoft Azure Government. It carries the license key, a hashed computer ID, and the application version, not your data, and it picks up proxy settings automatically. It runs every 7 days, and the application keeps working for up to 14 days if it cannot reach the service.
- **Disconnected networks.** STIGrevolution is supported on disconnected, classified, and air-gapped networks. Contact us and we will set up licensing for your environment.
- **Trial and license.** A 14-day free trial with no license key needed to start. A license runs 365 days from the date of purchase, and can be moved to another computer by returning it in the application and activating with the same key.

Purchase is subject to the STIG Solution Terms of Use and the STIGrevolution EULA, at stigsolution.com/terms-of-use/ and stigsolution.com/eula/stigrevolution/.

About STIG Solution

STIG Solution LLC builds tools for the people accountable for RMF packages on DoD and federal systems: STIGreviewer, a free replacement for DISA STIG Viewer; STIGrevolution, for working checklists at the scale of a whole package; and NESSviewer, for turning Nessus scan data into package artifacts. We also publish a STIG revisions file every quarter, free, at stigsolution.com/free-resources/.

Questions, a demonstration, or licensing: mail@stigsolution.com. Documentation is at stigsolution.com/docs/.

This paper describes STIGrevolution as of October 2026. STIG Solution is not affiliated with or endorsed by DISA or the U.S. Department of Defense. Nessus and ACAS are products of Tenable, Inc.; eMASS is a Department of Defense system. © 2026 STIG Solution LLC.