

REVIEW

Review: AttackIQ watches the watchers

This penetration testing tool is configured to operate from the inside, with the primary goal of identifying flaws, misconfigurations and outright shortcomings in all other cybersecurity defenses.

By **John Breeden II**

Nearly every cybersecurity program that has ever been reviewed by CSO has had one thing in common: its creators insist that they have the best product to watch over and protect network assets. Some are even interoperable with one another, so there is no reason not to have multiple defenses in place protecting networks. And yet, companies that have deployed multiple cybersecurity tools still get breached every day.

The problem might be one that Roman poet and satirist Juvenal pointed out back in the year 348 when he asked, “Quis custodiet ipsos custodes?” (“Who will guard the guards themselves?”) Juvenal’s point was that unless there is some kind of oversight, we only have the protectors’ word that they are always acting in our best interest.

In cybersecurity, AttackIQ was created to watch our watchers. It’s a penetration testing tool, but one that is configured to operate from the inside, with the primary goal of identifying flaws, misconfigurations and outright shortcomings in all other cybersecurity defenses. It can be used to pit various defenses against one another to see which works best for an environment, to discover areas where existing defenses unnecessarily overlap, or to identify bad configurations that are preventing security tools from properly operating.

The main AttackIQ management console either sits in the cloud or can be installed locally on premises if an organization wishes. In addition to the main console, which is used to configure and deploy tests against protected assets as well as collecting those results,

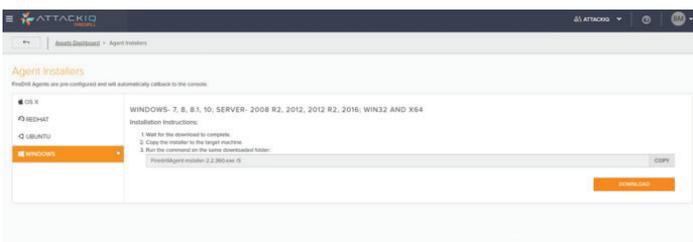


Getty Images

users will need to deploy agents. There are agents available for all forms of Windows and Mac OS systems, plus most flavors of Linux. Deploying those agents involves a fairly simple wizard-supported process to ensure that the right agents get to the correct assets.

There are actually two types of agents: static and dynamic. The static agents install onto an asset and remain there forever. They are perfect for critical assets that always need to be protected. The dynamic agents can be installed on systems for specific tests and then can be removed or moved to other systems. One such use would be to periodically test non-critical assets, such as antivirus protections on endpoints. Most deployments end up being about 80 percent static agents and 20 percent dynamic, according to AttackIQ officials. Pricing for AttackIQ is a tiered subscription model based on the number of agents used.

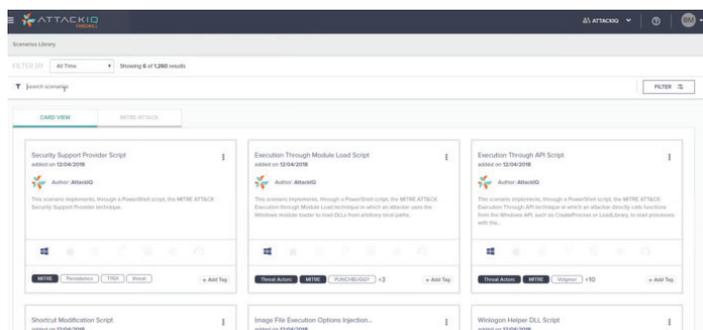
Once the agents are in place, users can choose from an existing library of 1,260 attack scenarios, all of which are highly configurable based on the unique environment where they are deployed. AttackIQ is constantly expanding its scenario library. Each scenario can be modified using a very easy-to-use wiz-



John Breeden II/IDG

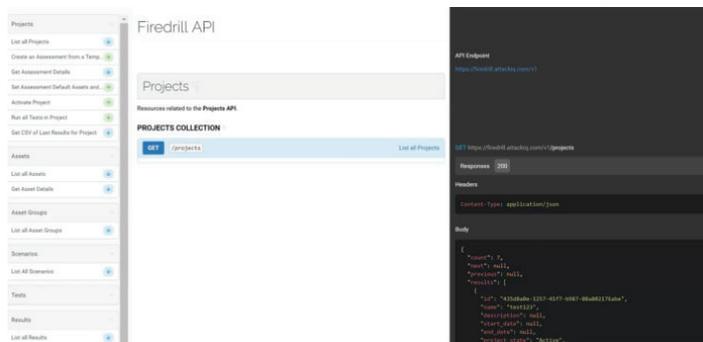
Because AttackIQ is operating from inside the network, agents need to be deployed on assets to complete tests, though some agents are considered mobile and can bounce around from asset to asset. AttackIQ supports all Windows and Mac OS configurations, as well as all flavors of Linux.

ard that ensures proper deployment, or if users feel comfortable doing so, the Python code they are written in can be edited directly. Users can additionally use the wizards to create their own scenarios with the existing toolset. And because AttackIQ is designed to run in working production environments, all attacks associated with a test have been defanged so they won't cause any damage.



John Breeden II/IDG
Defanged attack scenarios are grouped into modules for easy use and configuration before deployment. There are currently 1,260 scenarios as of the writing of this review, and the list is constantly expanding.

In our testing we first deployed a scenario involving credential scraping against assets in a testbed protected by some combination of antivirus for endpoints and more hardcore cybersecurity protections on core assets. Protections in some cases overlapped. When launched, scenarios wake up agents, deploy whatever instructions they need and then have the agents perform testing. Results are sent back to the management console, and the agent either goes back to sleep or moves on to the next scheduled test. You can set testing for individual scenarios to be continuous if you want, with tests repeating daily, weekly, monthly or based on whatever schedule works best for the environment.

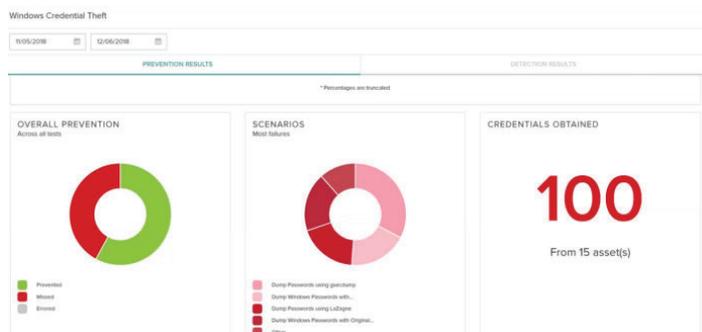


John Breeden II/IDG
Although the scenarios are designed to be modified using the wizard, they are all written in Python and can be modified directly, or new ones can be created to support unique environments. If users want to work directly in the APIs, AttackIQ will support it.

Testing can take some time to complete depending on the scope and type of test, but generally lasts a few minutes. Once

complete, users can see a general report showing which assets failed the testing and which passed. Clicking on those results explains which defenses are responsible for protecting assets, and why others failed to do so.

Over the course of testing with AttackIQ, it became obvious why a testing tool like this that is designed to evaluate other tools can be so helpful in defending modern networks. In one case we discovered that an antivirus program was blocking all attacks against endpoints, making a secondary defense that was also deployed on them largely unnecessary. In another scenario, an advanced cybersecurity program that should have been blocking attacks leveled at core assets was sometimes failing. The reason was that it had never been taken out of discovery mode on assets that were getting compromised, so it was monitoring the attacks but not actually preventing them.



John Breeden II/IDG
Like most testing type programs, AttackIQ gives a detailed look at how many assets failed various kinds of simulated attacks.

In still another scenario, multiple cybersecurity programs were interfering with one another, leading to decreased efficiency and openings for attackers to slip through. The list of problems and the reasons behind them were extremely interesting, and would have been difficult or impossible to discover without the dedicated AttackIQ tool.

AttackIQ can also be configured to interface with any network security information and event management system (SIEM). This can be helpful if defenses are set to report potentially suspicious behavior to a SIEM instead of taking any direct action. In that case, AttackIQ launches the scenario and then queries the SIEM to see if its specific attack behavior was reported at that time. Thereafter, the report from AttackIQ is identical to others. It shows whether an asset passed the test and why, and also what is to blame for any failures.

Not only will AttackIQ identify weak spots or flaws in existing defenses, but it will also find areas where misconfigurations or installation mistakes are preventing good cybersecurity tools from operating properly. In this era of incredibly complex networking where everything is a unique environment, AttackIQ can help ensure that the best defenses are in place and that they are operating at maximum efficiency. Who will watch over the cybersecurity watchers? AttackIQ can.