APPGUARD

REPORT

How Can You Achieve Zero Trust Endpoint Protection?

The point of applying zero trust within the endpoint is to reap better protection for less effort. Alternatives are ineffective and labor intensive because they monitor and investigate vast, diverse volumes of detection and indicator data from multiple perspectives at multiple stages of malware attacks; before and after compromise. They are parsing infinite possibilities, requiring more tools, more personnel, and more skills every year. Those who think machine learning will help them scale are finding that the single most pervasive characteristic in enterprise IT – **CHANGE** – is also machine learning's greatest adversary. A far different approach is needed.

AppGuard's Approach:

Rather than trying to scale to parse more, AppGuard's endpoint zero trust takes the opposite approach of drastically reducing what needs to be monitored and analyzed. It does this by avoiding the quagmire of telling "good" from "bad" and "normal" from "abnormal" by instead blocking those actions malware needs done. This replaces parsing infinity with suppressing hundreds of actions within an endpoint years of industry research have revealed are necessary for adversaries to attain their goals. With AppGuard, malware recognition is not required. Alternatives only succeed when they are able to recognize **every** piece of malware.

Malware's actions are performed by processes, which spring from applications, utilities, and untrustworthy files. Zero trust expects applications and utilities to go rogue at any moment. It uses containment, isolation, default-deny, and other controls to disrupt malware's intended actions. And it doesn't make "statistical guesses." If it cannot deterministically block, then it restrains. Sec-Ops is spared from the tug of war between false positives and false negatives.

All malware attacks have one thing in common. At least one app was involved in letting the malware in and/or doing the resulting harm.

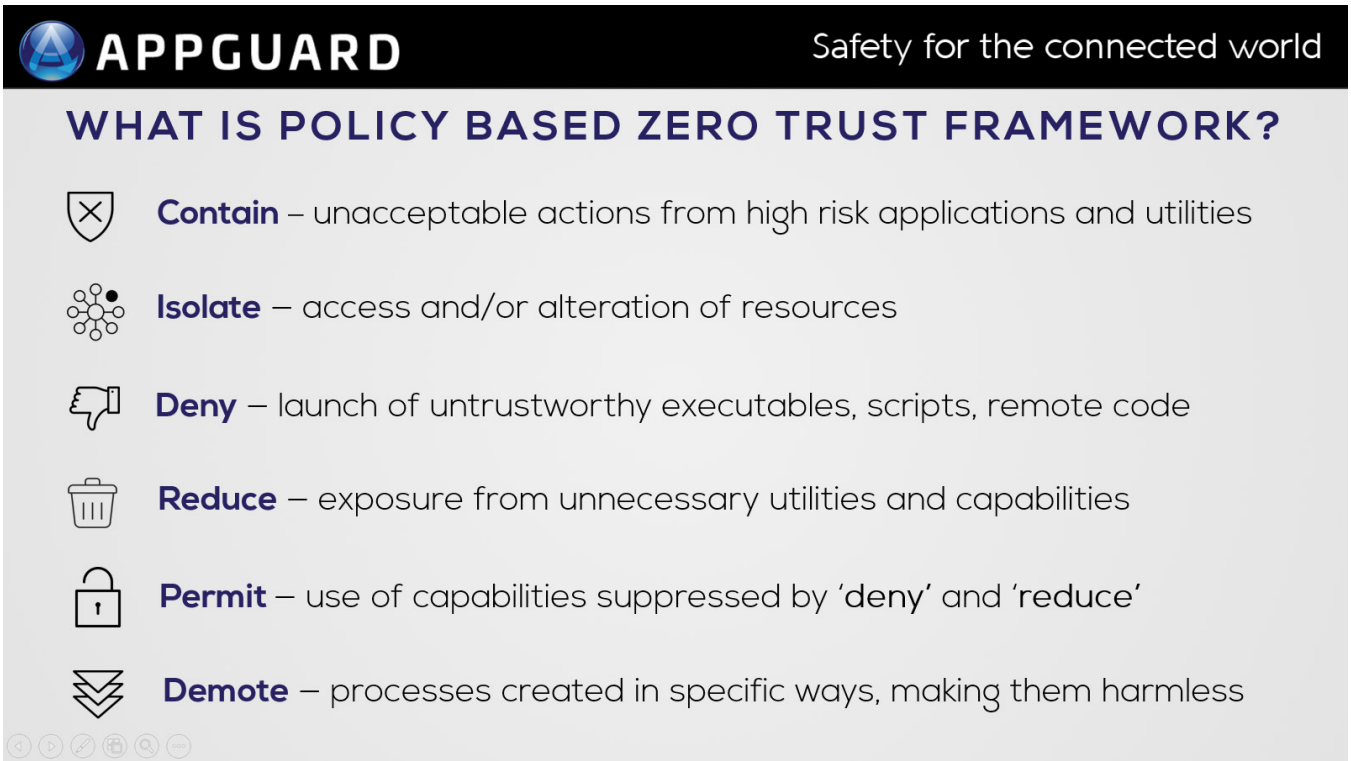
703.786.8884
sales@appguard.us
www.appguard.us

Failed conformance controls such as white-listing, HIPS, and sandboxing require too much endpoint state information that needs to be revised following changes such as application updates/patches. AppGuard's endpoint zero trust is based on patented higher abstractions that simplify policy formulation and automatically adapt to lifecycle changes. For example, app containment begins with its parent executable and automatically extends to any resulting process from the app's operation. This means very little state information is required for policy formulation, and updates/patches do not necessitate policy updates. Further, it accounts for the unanticipated.

Over 90% of enforced policies are defined by default. Agents typically run many months without policy updates – some have run for years. Containment is enforced uniformly to all at-risk apps, avoiding the app-specific policy quagmires of alternatives.

Customers praise AppGuard's real-time protection effectiveness and its near set-and-forget operations. Endpoint zero trust defeats malware without having to detect it, resulting in better protection and fewer operations. Further, other cyber defense layers see substantially lower alert volumes because malware attacks are stopped at endpoints in real time.

The Endpoint Zero Trust Framework



703.786.8884
sales@appguard.us
www.appguard.us

Use-Case	How Zero Trust Mitigates Risks & Accommodates Legitimate Usage
Unpatched App or Zero-Day Exploit	Does not allow an App or any process it spawns to install malware or steal/alter the memory of other App/OS processes. This alleviates patch/vulnerability management pressure. For AppGuard, containing an App is as simple as adding a song to a playlist, and it does not require adjustments later.
Drive-by Download	Scripts and executables are not allowed to launch unless proven trustworthy via validated digital signature or other means; those allowed to launch are not allowed to do harmful actions.
Server with Mission Critical App has mysterious, malicious process running	Any malware that somehow gets onto a server cannot read/write the memory, directories, executables, or data files of the 'isolated' mission critical App. IT/Sec-Ops can usually safely run the App until a maintenance window.
Pass the Hash/Ticket Attacks	Blocks credential thefts by granting access to trustworthy processes **only**. No IT/Sec-Ops actions are required; eliminates alerts that other tools would otherwise make.
Non-Malware Attacks	Prevents unauthorized actions by built-in tools, yet allows limited use by end-users and full-use by IT/Sec-Ops. This requires fewer than a dozen deployment-specific policy rules that rarely require adjustment later.
Code Injection Attacks	Blocks clearly untrustworthy App process changes and ensures the App's processes cannot do harmful actions in case they ever do run malicious code. Spares IT/Sec-Ops from the false-positive/negative quagmires of behavior analytics and other tools.
Remote Code Execution Attacks from other Endpoints	These built-in capabilities (e.g., Remote PowerShell, PsExec-like, SSH/shell, etc.) are locked/unlocked to ensure only IT/Sec-Ops can use them on demand, even if adversaries somehow steal elevated privilege credentials.

Contact Us

VA Office

14170 Newbrook Drive
Suite LL-01
Chantilly, VA 20151
USA

NY Office

333 Seventh Avenue
10th Floor
New York, NY 10001
USA

Tokyo Office

(Blue Planet-works, Inc.)
Daiwa Jingumae Bldg.,
3Fl2-4-11 Jingumae
Shibuya-Ku Tokyo,
150 - 0002
Japan

703.786.8884
sales@appguard.us
www.appguard.us