II.	**Multi-Platform Management**	
	Windows, Mac, and Linux machines must be managed from one management console.	
III.	**Updating Bandwidth Consumption**	
a.	Updating of endpoints should have the ability to set pre-configured available bandwidth used for both software updating and threat definition updates (e.g.,64,128, 256Kbps, etc.)	
b.	Must have the option to set up a local cache updating server within the on premise network environment to minimize large software engine update.	
c.	Must have an Update Management Policy that contains the configuration of update schedules on managed endpoints.	
d.	Virus signature updates must be under 30 KB	
IV.	**Deployment Options**	
	Deploying the endpoint agent must support the following methodology: 1) Email setup link 2)via AD Startup/Shutdown script 3)AD Login script 4) SCCM 5) Include the endpoint agent installation to a gold image	
V.	**SIEM Integration**	
	Must have the capability to extract events and alerts information	

	from the Cloud Dashboard to a local SIEM.	
VI.	**API for Endpoint Management**	
a.	Must have APIs offered as RESTful HTTP endpoints over the public internet.	
b.	APIs must have the capability to query tenants, enumerate and manage endpoints and servers, and query alerts and manage them programmatically.	
VII.	**Role Management**	
a.	Must have the capability to allow the separation of estate management to different administrator login.	
b.	Must provide admins the capability to assign predefined administrative roles to users who need access to the Admin Console.	
c.	Must be able to create custom roles and assign the products and access needed.	
VIII.	**Microsoft AD Synchronization**	
	Must have the capability to only allow outbound synchronization of Users/Groups from the local Active Directory servers to the Cloud Dashboard for policy management.	
IX.	**Microsoft Azure AD Authentication**	
a.	Must have the capability to log in to the Admin Dashboard and Self-Service Portal using Azure AD Login	
b.	Must have the capability to automatically login to the Admin	

	Dashboard/Self Service Portal if already authenticated in the web browser with Azure AD login from a different application/ service.	
X.	**Policies**	
a.	Selected policies should be able to be applied to either users or devices.	
b.	Policies must have the capability to be disabled automatically based on a scheduled time and date.	
XI.	**Enhanced Tamper Protection**	
a.	Must have the capability to prevent local administrative users or malicious processes from disabling the endpoint protection.	
b.	Must have the capability to prevent the following actions on the endpoint protection solution: 1) Stopping services from the Services Ul 2) Kill services from the Task Manager Ul 3) Change Service Configuration from the Services Ul 4) Stop Services/edit service configuration from the command line 5) Uninstall 6) Reinstall 7) Kill processes from the Task Manager Ul (desired) 8) Delete or modify protected files or folders	

	9) Delete or modify protected registry keys	
c.	Must be able to export Tamper Protection passwords in CSV or PDF formats.	
XII.	Threat Protection	
a.	Must protect against multiple threats, both known and unknown, and provide a trusted and integrated approach to threat management at the endpoint.	
b.	Must protect endpoint systems against viruses, spyware, Trojans, rootkits, and worms on workstations and laptops regardless of their nature or the concealment mechanisms used.	
c.	Must protect against threats related to executable files, as well as document files containing active elements such as macros or scripts. lt must protect against exploits resulting from discovery (whether published or not) of security flaws in systems or software.	
d.	Must have the capability to 'lookup' files in real-time to verify if they are malicious. This feature checks suspicious files against the latest malware in the vendor's Threat Intelligence database in the cloud.	
e.	Must have the capability to do real-time scanning of local files and network shares the moment the user tries to access them. Access must be denied unless the file is healthy.	
f.	Must have the capability to do real-time scanning of end-users Internet Access. It must monitor and classify the Internet websites	

	according to their level of risk, and make this technology available to endpoint systems. A site known to host malicious code or phishing sites must be proactively blocked by the solution to prevent any risk of infection or attack against a flaw of the browser used. The solution must carry out checks against a database of compromised websites that are constantly being updated with new sites identified per day.	
g.	Must protect managed systems from malicious websites in real-time, whether end-users work within the company or outside the company's secure network - at home or through public Wi-Fi. All browsers on the market must be supported (Internet Explorer, Firefox, Safari, Opera, Chrome, etc.)	
XIII.	**Anti-rootkit Detection**	
	Must identify a rootkit when reviewing an element without overloading the endpoint system. Rootkits must be proactively detected.	
XIV.	**Suspicious Behavior Detection**	
a.	Must be able to protect against unidentified viruses and suspicious behavior.	
b.	Must have both pre-execution behavior analysis and runtime behavior analysis.	
c.	Must be able to identify and block malicious programs before execution.	
d.	Must be able to dynamically analyze the behavior of programs running on the system and detect	

	then block activity that appears to be malicious. This may include changes to the registry that could allow a virus to run automatically when the computer is restarted.	
e.	Must provide protection against buffer overflow attacks.	
XV.	**Scanning**	
a.	Must provide a scheduled scanner to run depending on the selected frequency or by manually triggering through Windows Explorer to scan the specified directories (local, remote or removable), with analysis parameters used, which may be different from the ones selected for real-time protection.	
b.	Must have the capability to scan archives such as zip, cab, etc. which can be enabled via policy settings.	
XVI.	**Advanced Deep Learning mechanism**	
a.	The system shall be light speed scanning; within 20 milliseconds, the model shall able to extract millions of features from a file, conduct deep analysis, and determine if a file is benign or malicious. This entire process happens before the file executes.	
b.	Must be able to prevent both known and never-seen-before malware, likewise must be able to block malware before it executes.	
c.	Must protect the system even with offline and will not rely on signatures.	
d.	Must classify files as malicious, potentially unwanted apps (PUA) or benign. Deep learning must	

	also focus on Windows portable executables.	
e.	Able to perform new Zero day's threat scanning offline (without internet).	
f.	Must be Smarter - should be able to process data through multiple analysis layers, each layer making the model considerably more powerful.	
g.	Must be scalable - should be able to process significantly more input, can accurately predict threats while continuing to stay up-to-date.	
h.	Must Lighter - model footprint shall be incredibly small, less than 20MB on the endpoint, with almost zero impact on performance.	
i.	The deep learning model shall be trail and evaluate models end-to-end using advanced developed packages like Keras, Tensorflow, and Scikit-learn.	
XVII.	**Exploit Prevention/Mitigation must detect and stop the following known exploits:**	
	1,) Enforcement of Data Execution Protection (DEP) Prevents abuse of buffer overflows 2) Mandatory Address Space Layout Randomization Prevents predictable code locations (ASLR) 3) Bottom-up ASLR	

	Improved code location randomization 4) Null Page (Null Dereference Protection) Stops exploits that jump via page 0 5) Heap Spray Allocation Reserving or pre-allocating commonly used memory addresses, so they cannot be used to house payloads. 6) Dynamic Heap Spray Stops attacks that spray suspicious sequences on the heap 7)Stack Pivot Stops abuse of the stack pointer 8) Stack Exec (MemProt) Stops attacker's code on the stack 9) Stack-based ROP Mitigations (Caller) Stops standard Return-Oriented Programming attacks 10) Branch-based ROP Mitigations (Hardware Augmented) Stops advanced Return-Oriented Programming attacks 11) Structured Exception Handler Overwrite Protection (SEHOp) Stops abuse of the exception handler 12) Import Address Table Access Filtering (lAF) (Hardware Augmented)	

	Stops attackers that lookup API addresses in the IAT 13) LoadLibrary API calls Prevents loading of libraries from UNC paths 1.4) Reflective DLL Injection Prevents loading of a library from memory into a host process 15) Shellcode monitoring Detecting the adversarial deployment of shellcode involves multiple techniques to address things fragmented shellcode, encrypted payloads, and null free encoding 16)VBScript God Mode Have the ability to detect the manipulating of the safe mode flag on VBScript in the web browser 17)WoW64 Must have the ability to prohibit the program code from directly switching from 32-bit to 64-bit mode (e.9., using ROP) while still enabling the WoW54 layer to perform this transition. 18) Syscall Stops attackers that attempt to bypass security hooks 19) Hollow Process Protection Stops attacks that use legitimate processes to hide hostile code 20) DLL Hijacking	

	Gives priority to system libraries for downloaded applications 21) Application Lockdown Will automatically terminate a protected application based on its behavior; for example, when an office application is leveraged to launch PowerShell, access the WMI, run a macro to install arbitrary code or manipulate critical system areas; the solution must block the malicious action - even when the attack doesn't spawn a child process. 22) Java Lockdown Prevents attacks that abuse Java to launch Windows executables 23) Squiblydoo AppLocker Bypass Prevents regsvr32 from running remote scripts and code 24) CVE-2OL3-5331 & CVE-2014-4113 via Metasploit ln-memory payloads: Meterpreter & Mimikatz	
XVIII.	**Advanced Exploit Mitigation**	
	Must be able to protect against a range of exploits or "active adversary" threats such as the following: 1) Credential Theft Theft of passwords and hash information from memory, registry, or hard disk. 2) APC Violation	

	Attacks using Application Procedure Calls (APC) to run malicious codes. 3) Privilege Escalation Attacks escalating a low-privilege process to higher privileges to access systems. 4) Code Cave Utilization Malicious code that's been inserted into another, legitimate application. 5) Application Verifier Exploits Attacks that exploit the application verifier in order to run unauthorized software at startup.	
XIX.	**Malicious Traffic Detection (MTD)**	
	Must be able to detect communications between endpoint computers and command and control servers involved in a botnet or other malware attacks.	
XX.	**Intrusion Prevention System (IPS)**	
a.	Must be able to prevent malicious network traffic with packet inspection (IPS).	
b.	Must be able to scan traffic at the lowest level and block threats before harming the operating system or applications.	
XXI.	**Anti-Ransomware Protection**	
a.	Must have the ability for the encrypted files to be rolled back to a pre-encrypted state.	
b.	Both Anti-Exploit and Ransomware protection does not	

	need to have a Cloud Lookup to perform the detection.	
c.	The anti-crypto function shall look back at all the malicious file modifications made by that process and restores them to their original location.	
d.	Should a ransomware infection managed to get in, detailed historical tracking of where the infection originated and how it propagated will be reported courtesy of the Threat Cases (RCA).	
e.	Must be able to protect from ransomware that encrypts the master boot record and from attacks that wipe the hard disk.	
XXII.	**AMSI protection**	
a.	Must be able to protect against malicious code (for example, PowerShell scripts) using the Microsoft Antimalware Scan Interface (AMSI).	
b.	Must be able to scan code forwarded via AMSI before it runs, and the applications used to run the code are notified of threats. If a threat is detected, an event is logged.	
XXIII.	**Data Loss Prevention (DLP)**	
a.	Must be able to monitor and restrict the transfer of files containing sensitive data.	
b.	Must have the capability to create custom DLP policies or policies from templates.	
c.	Must have DLP policy templates that cover standard data protection for different regions.	

XXIV.	**Peripheral Control**	
a.	Must have the capability to control and restrict removable mass storage devices (USB sticks, CD Rom, USB external hard drives, iPods, MP3 players, etc.), as well as connection devices (Wi-Fi, Bluetooth, Infrared, Modems, etc.).	
b.	Must have the capability to add device exemptions either by Model ID or Instance ID.	
XXV.	**Application Control**	
a.	Must have the capability to limit the applications needed for specific user groups.	
b.	Must be able to detect and block application categories that may not be suitable for use in an enterprise environment.	
c.	Must have application categories for commonly used applications.	
XXVI.	**Web Control**	
a.	Must be able to block risky downloads, protect against data loss, prevent users from accessing web sites that are inappropriate for work, and generate logs of blocked visited sites.	
b.	Must have security options to configure access to ads, uncategorized sites, or dangerous downloads.	
c.	Must provide the administrator the ability to define "acceptable web usage" settings (defined by categories) in order to control the sites on which users are allowed to visit. Admin must have control access to websites that have been	

	identified and classified in their own categories.	
d.	Must have a data loss protection option that allows the administrator to control access to web-based email and file downloads, with choices of blocking the data, allowing data sharing, or customizing this choice.	
XXVII.	**Windows Firewall Policy**	
a.	Must be able to monitor and configure Windows Firewall on managed computers and servers using a Windows Firewall policy.	
b.	Must be able to apply the Windows Firewall policy to individual devices (computers or servers) or groups of devices.	
XXVIII.	**Endpoint Detection and Response**	
a.	Must have the capability to identify what happened, where a breach originated, what files were impacted, and provides guidance on how to strengthen an organization's security posture	
b.	Must be able to record chain of events that occurred after an infection has been detected, enabling you to determine the origin of the infection, any resulting damage to assets, potentially exposed data, and the chain of events leading up to the halting of the infection.	
c.	Shall provide a summary of the event: What the exploit was discovered, where the beacon event occurred (an asset), when it occurred, how the infection succeeded. Eg. "Outlook.exe."	

d.	Shall provide recommendations to address the problem: Things to look for post-attack. Eg. Aside from files being restored from encrypted ones, check browser settings to ensure no vulnerabilities were created as a result of the infections.	
e.	Activity Record allows administrators to add notes to the case. All case-related notes will be listed in this column.	
f.	There are also buttons to enable the admin to modify the status of the case (New, ln progress, and Closed) and to set priority (Low, Medium, High). When closing, the administrator can add notes and is also required to confirm (via checkboxes) that remediation steps were taken: analyzed impact on files/assets and relevant environmental improvements were implemented.	
g.	Shall provide a tabular view of everything affected during the attack. Items can be filtered based on type - e.g., files, processes, registry keys. The administrator can view information about each item, e.g., Filename (victim file or malware agent), process ID, start/stop timestamp of the event.	
h.	Shall indicate the beginning of the root cause, charting out the series of events resulting from the attack as a collection of nodes. Each node contains specific information about files, processes, registry keys, etc. involved at that stage. The beacon event (marked with a blue dot) will be identified in the chain, but any events executed by the	