303 BIG-IP ASM SPECIALIST
EXAM BLUEPRINT

ABOUT THE 303 BIG-IP ASM SPECIALIST EXAM.

The BIG-IP ASM Specialist exam identifies individuals who are qualified to design, implement, and maintain ASM, including advanced features. They will likely be a senior network, system, and/or application security engineer with at least one year of relevant job experience responsible for delivering highly available, scalable, and secure applications with the ASM technology. The BIG-IP ASM Specialist understands the underlying principles of ASM and can draw on that insight to integrate ASM with other platforms and products. This exam is based on **TMOS v12.1**.

WHAT IS THE 303 BIG-IP ASM SPECIALIST EXAM BLUEPRINT?

F5 Certified! Exam Blueprints list all the objectives an exam has to measure, much like a syllabus for the exam itself. The blueprint provides the detailed breakdown of the skills and knowledge a candidate should have to pass the exam. Blueprints can be used to identify areas for additional study, and are best used in conjunction with the Exam Study Guides.

PREREQUISITE:

F5 Certified BIG-IP Administrator (F5-CA)

THIS EXAM IS BASED ON V12.1

f5 CERTIFIED® TECHNOLOGY SPECIALIST

Section 1:	Architecture/Design and Policy Creation	Cognitive Complexity
Objective 1.01	Explain the potential effects of common attacks on web applications	U/A
Examples	Understand and describe how the ASM can affect clients and applications directly while in either transparent or blocking mode Summarize the OWASP Top Ten	
Objective 1.02	Explain how specific security policies mitigate various web application attacks	U/A
Example	Understand/interpret an iRule or LTM policy to map application traffic to an ASM policy Explain the trade-offs between security, manageability, false positives, and performance	
Objective 1.03	Determine the appropriate policy features and granularity for a given set of requirements	A/E
Example	Understand application (security) requirements and convert requirements to technical tasks	
Objective 1.04	Determine which deployment method is most appropriate for a given set of requirements	A/E
Examples	Determine which deployment method is most appropriate given the circumstances (web services, vulnerability scanner, templates, rapid deployment model)	
Objective 1.05	Explain the automatic policy builder lifecycle	U/A
Examples	Create any profiles required to support the policy deployment (xml, JSON, logging profiles) Implement anomaly detection appropriate to the web app (D/DoS protection, brute force attack, web scraping, proactive bot defense)	
Objective 1.06	Review and evaluate policy settings based on information gathered from ASM (attack signatures, DataGuard, entities)	A/E
Examples	Configure initial policy building settings (automatic policy builder settings)	
Objective 1.07	Define appropriate policy structure for policy elements	U/A
Example	Define appropriate policy structure for policy elements (URLs, parameters, file types, headers, sessions & logins, content profiles, CSRF protection, anomaly detection, DataGuard, proactive bot defense)	
Objective 1.08	Explain options and potential results within the deployment wizard	U/A
Example	Describe options within the deployment wizard (deployment method, attack signatures, virtual server, learning method Select the appropriate ASM deployment model given the business requirements	

Cognitive Complexity Key:
R=Remember
A/E=Analyze/Evaluate
U/A=Understand/Apply

Objective 1.09	Explain available logging options	R
Example	Explain the specifications of the remote logger (ports, types of logs, formats, address)	
Objective 1.10	Describe the management of the attack signature lifecycle and select the appropriate attack signatures or signature sets	U/A
Example	Understand management of attack signature lifecycle (staging, enforcement readiness period) and select appropriate attack signatures or signature sets.	
Section 2:	**Policy Maintenance and Optimization**	Cognitive Complexity
Objective 2.01	Evaluate the implications of changes in the policy to the security and functionality of the application	A/E
Examples	Evaluate whether the rules are being implemented effectively and appropriately to meet security and/or compliance requirements and make changes as appropriate	
Objective 2.02	Explain the process to integrate natively supported third party vulnerability scan output and generic formats with ASM	U/A
Examples	Refine appropriate policy structure for policy elements (URLs, parameters, file types, headers, sessions & logins, content profiles, CSRF protection, anomaly protection). Explain how to manage policies using import, export, merge, and revert	
Objective 2.03	Evaluate whether rules are being implemented effectively and appropriately to mitigate violations	A/E
Examples	Evaluate the implications of changes in the policy to the security and vulnerabilities of the application	
Objective 2.04	Determine how a policy should be adjusted based upon available data	A/E
Examples	Tune an ASM policy for better performance, including use of wildcards to improve efficiency	
Objective 2.05	Define the ASM policy management functions	R
Examples	Identify the status of the policy Define the violation types that exist in ASM Describe how to merge and differentiate between policies	

Cognitive Complexity Key:
R=Remember
A/E=Analyze/Evaluate
U/A=Understand/Apply

Section 3:	Review Event Logs and Mitigate Attacks	Cognitive Complexity
Objective 3.01	Interpret log entries and identify opportunities to refine the policy	A/E
Examples	Examine traffic violations, determine if any attack traffic was permitted through the ASM and modify the policy to remove false positives Locate and interpret reported security violations by end users and application developers	
Objective 3.02	Given an ASM report, identify trends in support of security objectives.	U/A
Examples	Understand and describe each major violation category and how ASM detects common exploits Generate reporting for the ASM system and review the contents of the reports (anomaly statistics, charts, requests, PCI compliance status)	
Objective 3.03	Determine the appropriate mitigation for a given attack or vulnerability	A/E
Examples	Take appropriate action on reported security violations by end users and application developers Modify ASM policy to adapt to attacks	
Objective 3.04	Decide the appropriate method for determining the success of attack mitigation	A/E
Examples	Choose an appropriate user defined attack signature to respond to particular traffic	

Section 4:	Troubleshoot	Cognitive Complexity
Objective 4.01	Evaluate ASM policy performance issues and determine appropriate mitigation strategies	A/E
Examples	Analyze performance graphs and statistics along with ASM configurations to determine the root cause of performance issues and appropriate remediation to the configuration based on Guaranteed Logging	
Objective 4.02	Understand the impact of learning, alarm, and blocking settings on traffic enforcement	U/A
Examples	Ensure that the security policy is inspecting web application traffic (application is functional and the policies are parsing the traffic)	
Objective 4.03	Examine policy objects to determine why traffic is or is not generating violations	A/E
Examples	Examine Security Event Logs and ASM configurations to determine expected violations based on the logging profile assigned to the virtual server	

Cognitive Complexity Key:
R=Remember
A/E=Analyze/Evaluate
U/A=Understand/Apply

Objective 4.04	Identify and interpret ASM performance metrics	U/A
Examples	Understand the impact of ASM iRules on performance. Understand the impact of traffic spikes on ASM performance and available mitigation strategies	
Objective 4.05	Evaluate ASM system performance issues and determine appropriate mitigation strategies	A/E
Examples	Correlate performance issues with ASM policy changes based on security policy history information and system performance graphs	
Objective 4.06	Recognize ASM specific user roles and their permissions	R
Examples	Recognize differences between user roles/permissions Recognize ASM specific user roles	

Cognitive Complexity Key:
R=Remember
A/E=Analyze/Evaluate
U/A=Understand/Apply

Cognitive Complexity Descriptions

Lower Order Thinking Skills → Higher Order Thinking Skills

Remember	Understand/Apply	Analyze/Evaluate	Create
Information retrieval	Knowledge transfer	Critical thinking and reasoning	Innovation or Creative thinking
Rote memorization	Comprehension or Ability to apply knowledge to a standard process	Determine how parts relate to whole or Knowledge integration and application to new situation(s)	Forming an original work product
Retrieve relevant knowledge from long-term memory	Construct meaning from information	Make judgments based on criteria	Combine or reorganize parts to form a new pattern or structure
e.g., recall, retrieve, recognize	e.g., interpret, classify, compare, explain, implement	e.g., troubleshoot, attribute, diagnose, critique	e.g., generate, plan, produce

Alpine Testing Solutions' suggested cognitive complexity levels and associated verb references consider multiple approaches to defining cognitive processing (e.g., Anderson et al., Webb, Bloom, Frisbie). Above material created with assistance from Alpine and distributed with Alpine's permission as an attachment to certification test blueprints.

Alpine Testing Solutions, Inc. (Alpine) gives F5 Networks permission to distribute the PDF "Cognitive Complexity Description 20130418.pdf" as an attachment to certification test blueprints created with assistance from Alpine into the exam blueprint.