



# CompTIA AutoOps+ Certification Exam Objectives

**EXAM NUMBER: AT0-001 V1**

# About the Exam

The CompTIA AutoOps+ AT0-001 V1 certification exam will certify the successful candidate has the knowledge and skills required to:

- Use coding techniques and best practices to implement automation throughout the enterprise.
- Facilitate innovative approaches to system configuration and provisioning.
- Use techniques to manage and troubleshoot continuous integration and continuous delivery (CI/CD) pipelines.
- Adhere to security best practices.

This is equivalent to at least 2–3 years of experience in a core IT operations role, such as a network, cloud, or systems administrator. Recommended certifications are Network+, Linux+, Cloud+, Server+, or similar certifications.

These content examples are meant to clarify the test objectives and should not be construed as a comprehensive listing of all the content of this examination.

## EXAM ACCREDITATION

TBD

## EXAM DEVELOPMENT

CompTIA exams result from subject matter expert workshops and industry-wide survey results regarding the skills and knowledge required of an IT professional.

## CompTIA AUTHORIZED MATERIALS USE POLICY

CompTIA Certifications, LLC is not affiliated with and does not authorize, endorse, or condone utilizing any content provided by unauthorized third-party training sites (aka “brain dumps”). Individuals who utilize such materials in preparation for any CompTIA examination will have their certifications revoked and be suspended from future testing in accordance with the CompTIA Candidate Agreement. In an effort to more clearly communicate CompTIA’s exam policies on use of unauthorized study materials, CompTIA directs all certification candidates to the [CompTIA Certification Exam Policies](#). Please review all CompTIA policies before beginning the study process for any CompTIA exam. Candidates will be required to abide by the [CompTIA Candidate Agreement](#). If a candidate has a question as to whether study materials are considered unauthorized (aka “brain dumps”), they should contact CompTIA at [examsecurity@comptia.org](mailto:examsecurity@comptia.org) to confirm.

## PLEASE NOTE

The lists of examples provided in bulleted format are not exhaustive lists. Other examples of technologies, processes, or tasks pertaining to each objective may also be included on the exam, although not listed or covered in this objectives document. CompTIA is constantly reviewing the content of our exams and updating test questions to be sure our exams are current, and the security of the questions is protected. When necessary, we will publish updated exams based on existing exam objectives. Please know that all related exam preparation materials will still be valid.

## TEST DETAILS

Required exam	AutoOps+ AT0-001 V1
Number of questions	Maximum of 60
Types of questions	Multiple-choice and performance-based
Length of test	60 minutes
Recommended experience	2–3 years of experience in a core IT operations role, such as a network, cloud, or systems administrator. Recommended certifications are Network+, Linux+, Cloud+, Server+, or similar certifications.
Passing score	600 (on a scale of 100–900)

## EXAM OBJECTIVES (DOMAINS)

The table below lists the domains measured by this examination and the extent to which they are represented.

DOMAIN	PERCENTAGE OF EXAMINATION
1.0 Automation Coding Concepts	31%
2.0 System Configuration	25%
3.0 Continuous Integration	24%
4.0 Continuous Delivery	20%
<b>Total</b>	<b>100%</b>

# 1.0 Automation Coding Concepts

## 1.1 Given a scenario, use code to support automation.

- Variables
  - Scope
    - ◆ Global
    - ◆ Local
- Conditional statements
- Iterations
- Parameters
- Functions
- Application logs
- Regular expression (regex)
- Data types
  - Primitive
  - JavaScript Object Notation (JSON)
  - YAML Ain't Markup Language (YAML)
- Command-line tools
  - `grep`
  - `awk`
  - `sed`
  - `jq`
  - `Cmdlets`
- Dependency management
  - `Dockerfile`
  - `requirements.txt`
  - `Packages`
  - `Libraries`

## 1.2 Given a scenario, use source control techniques to version code.

- Git commands
  - Remote
    - ◆ `git fetch`
    - ◆ `git pull`
    - ◆ `git push`
  - Local
    - ◆ `git add`
    - ◆ `git commit`
  - `config`
- Semantic versioning
  - Pre-release versions
  - Major vs. minor releases
  - Filtering techniques
- Branching strategies
  - Release branching
  - Feature branching
- Branch naming conventions
- Commit life cycle
  - Hooks
  - Formatting
  - Linting
  - Detecting sensitive data
  - Comments

### 1.3 Explain concepts related to infrastructure as code (IaC).

- Reusability
- Immutability
- Declarative vs. imperative
- Idempotency

### 1.4 Given a scenario, troubleshoot common issues with the code life cycle.

- Syntax errors
  - Undefined variable errors
- Runtime errors
- Git errors
  - Merge conflicts
  - Authentication issues
  - Detached HEAD

## 2.0 System Configuration

### 2.1 Given a scenario, use configuration management techniques to deploy a solution.

- Drift detection and remediation
- State management
- Workload configuration
  - Certificates
  - Access control lists (ACLs)
- Virtual network infrastructure configuration
  - Subnets
  - Routers
  - Load balancers
  - Firewalls
  - Route tables
  - Domain Name System (DNS)

### 2.2 Compare and contrast basic approaches to automation.

- Remote vs. local
  - Remote
    - ◆ Windows Remote Management (WinRM)
    - ◆ Secure Shell (SSH)
    - ◆ Agent-based
  - Local
    - ◆ Command-line script
    - ◆ Background/foreground
- Declarative vs. imperative
  - Migrations
  - Ephemeral environments
  - Persistent environments
  - Script writing
- Provisioning techniques
  - Push
  - Pull
- Cloud-based events
  - Notification-based events
  - Messaging queues
  - Function as a service (FaaS)

### 2.3 Given a scenario, interact with Representational State Transfer–compliant (RESTful) systems to perform create, read, update, and delete (CRUD) operations.

- Headers
- Methods
  - GET
  - POST
  - DELETE
  - PATCH
  - PUT
- REST application programming interface (API) tools
  - curl
  - wget
  - Postman
- Response codes
  - 2XX
  - 3XX
  - 4XX
  - 5XX

## 2.4 Given a scenario, troubleshoot common issues with system configuration.

- API communication failures
  - Agent communication issues
- Application failures
- Certificate issues
- Syntax errors specific to configuration files
  - INI
  - YAML
  - JSON

# 3.0 Continuous Integration

## 3.1 Explain environmental factors related to continuous integration (CI) management.

- Secrets management
  - Encrypted key vault
  - Revocation
  - Lease management and expiration
  - Tokenization
  - Zero Trust
  - Dynamic secret rotation
- Artifact management
  - Artifact registry
  - Authorization
  - Authentication
  - Versioning
  - Immutability
  - External artifact repository
  - Software bill of materials (SBOM)
- Task runners
  - Environment provisioning

## 3.2 Explain workflow management concepts in CI.

- Orchestration techniques
  - Parallel vs. sequential
  - Step dependencies
  - Conditional execution
  - Automated rollbacks
  - Failure handling
- Pipeline-enhancing concepts
  - Security scanning
  - Containerization
  - Code quality scanning
  - Artificial intelligence (AI)-based log analysis

## 3.3 Given a scenario, analyze configurations to manage basic automation pipelines.

- Hooks and triggers
  - Webhooks
  - Chat-based operations (ChatOps)
- Pipeline triggers
  - Push trigger
  - Manual trigger
  - Scheduled trigger
  - Git variables
    - ◆ Tag
    - ◆ Branch
    - ◆ Commit
- Pipeline definition
  - Jenkinsfile
  - gitlab-ci.yml
  - github-actions.yml

# 4.0 Continuous Delivery

## 4.1 Given a scenario, implement techniques of continuous delivery (CD).

- Delivery methods
  - Canary
  - Blue-green
  - Rolling
  - In-place
- Quality assurance (QA) testing
  - Load testing
  - Regression testing
  - Integration testing
- Validation and remediation
  - Smoke testing
  - Post-deployment testing
  - Hotfixes

## 4.2 Explain concepts related to application service levels.

- Uptime
- Service-level objectives (SLOs)
- Service-level agreements (SLAs)
- Mean time to repair
- Mean time between failures (MTBF)
- Feedback loop

## 4.3 Compare and contrast different methods to secure a connection to providers.

- Command-line interface (CLI) configuration
- Software development kit (SDK)
- Identity and Access Management (IAM)
  - Machine identities
  - Accessing external APIs

# CompTIA AutoOps+ Acronym List

The following is a list of acronyms that appear on the CompTIA AutoOps+ AT0-001 V1 certification exam. Candidates are encouraged to review the complete list and attain a working knowledge of all listed acronyms as part of a comprehensive exam preparation program.

<b>ACRONYM</b>	<b>DEFINITION</b>
ACL	Access Control List
AI	Artificial Intelligence
API	Application Programming Interface
AWS	Amazon Web Services
CD	Continuous Delivery
CI	Continuous Integration
CI/CD	Continuous Integration and Continuous Delivery
CLI	Command-line Interface
CRUD	Create, Read, Update, and Delete
DNS	Domain Name System
FaaS	Function as a Service
GUI	Graphical User Interface
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IaC	Infrastructure as Code
IAM	Identity and Access Management
JSON	JavaScript Object Notation
MTBF	Mean Time Between Failures
OS	Operating System
POST	Power-on Self-test
QA	Quality Assurance
REST	Representational State Transfer
RTO	Recovery Time Objective
SBOM	Software Bill of Materials
SDK	Software Development Kit
SLA	Service-level Agreement
SLI	Service-level Indicator
SLO	Service-level Objective
SSH	Secure Shell
URL	Uniform Resource Locator
VM	Virtual Machine
WinRM	Windows Remote Management
YAML	YAML Ain't Markup Language

# CompTIA AutoOps+ Hardware and Software List

CompTIA has included this sample list of hardware and software to assist candidates as they prepare for the AutoOps+ AT0-001 V1 certification exam. This list may also be helpful for training companies that wish to create a lab component for their training offering. The bulleted lists below each topic are sample lists and are not exhaustive.

## SOFTWARE

- Bash
- PowerShell
- Python
- Git
- Configuration management tool
  - SaltStack
  - Chef
  - Puppet
- CI/CD tools
  - Jenkins
  - GitHub Actions
  - GitLab
  - Argo CD
  - Azure DevOps
- IaC tools
  - Terraform
  - CloudFormation
  - Bicep
- Containerization platform
  - Docker
- Virtualization platform
  - VMware
- REST clients
  - curl
  - Postman

## OTHER

- Public cloud provider subscription
- Cloud provider client
  - Amazon Web Services (AWS)