

UNCLASSIFIED

This document is classified at the **UNCLASSIFIED** level until information is added. It may then contain elements of Controlled Unclassified Information (CUI) or information at a higher classification level than the original classification of this document. POC: USCYBERCOM J72 CMF Branch, email: USCC_J721@cybercom.ic.gov.

UNITED STATES CYBER COMMAND



(U) Job Qualification Record (JQR) Cyber Mission Force

Basic Cyberspace Capability Developer

Version 1.0

CLASSIFICATION INSTRUCTIONS

This document's **first page** is **UNCLASSIFIED** when separated from the rest of this document. This document contains portion markings classified up to the **UNCLASSIFIED** level.

UNCLASSIFIED

(U) Release History

VERSION/ RELEASE	DATE	AUTHOR(S)	DESCRIPTION
1.0	13 Jul 22	AFCYBER (USCC JCL)	Initial Version

(U) Summary of Changes

VERSION/ RELEASE	DATE	AUTHOR(S)	REASON	DESCRIPTION

(U) Overview

(U) Job Qualification Record (JQR) Program: The JQR program provides commanders a consistent format to track individual training progress and readiness. The JQR documents formal and informal training options that can be leveraged to prepare an individual to perform in a specific work role or position. Job Qualification Standard (JQS) line items outline the actions to be demonstrated through various means (application, verbal, written) to measure an individual's ability to perform the Knowledge, Skills and Abilities (KSAs) of assigned tasks.

(U) The Joint Force Trainer (JFT), USCYBERCOM J7 will advise the Services when updates or changes are made to the standards that may affect the JQR. The Joint Cyber Training and Certification Standards (JCT&CS) follows an annual review and update cycle, and updated portions of the standards are communicated to the Services. The DoD Cyber Workforce Framework (DCWF) also has periodic updates; these will also be communicated to the Services.

(U) The Joint Curriculum Lead (JCL) has the responsibility to update or change to their JCL assigned work role JQR and ensure they comply with any new or changed joint requirements.

(U) This JQR shall be completed in accordance with and within timeframes specified by Cyberspace National Mission Force (CNMF), Service Cyber Components (SCC), and/or local unit commanders.

Cyber Mission Force JQR

Basic Cyberspace Capability Developer

(U) Introduction to the JQR: Cyberspace Capability Developer

(U) Purpose

(U) The purpose of this Job Qualification Record (JQR) is to communicate the individual level tasks, knowledge, skills and abilities necessary to perform the **Cyberspace Capability Developer** work role for all assigned personnel within the Combat Support Teams (CST) and National Support Teams (NST) who are assigned to Cyber Mission Force.

(U) This JQR serves two primary audiences:

- **(U) Team Leads:** use this JQR to identify the required tasks and KSAs required of a team member performing this work role, and
- **(U) Training Developers:** use the tasks and KSAs to inform the design and development of instruction.

(U) Contents

(U) This JQR contains 8 Tabs:

- (U) Tab 1- (Mandatory) Work Role description (USCYBERCOM Work Role Descriptions). Work Roles and positions may or may not be the same. Many positions can be encompassed into one work role training pipeline but a position is unique to itself.
- (U) Tab 2- (Mandatory) USCYBERCOM approved Work Role Tasks and KSAs at the Basic proficiency level as outlined in the Joint Cyber Training and Certification Standards (JCT&CS).
- (U) Tab 3- (Mandatory) Formal training requirements (training courses or curriculum) as outlined on the USCYBERCOM Training Pipeline.
- (U) Tab 4- (Mandatory) JQS line items linked to approved Basic proficiency level Tasks and KSAs.
- (U) Tab 5- (As Required) Service Specific/Operational Training Requirements (SCC, JFHQ-C, JFHQ-DoDIN, CNMF).
- (U) Tab 6- (Mandatory) Record of assessments, qualifications, Trainer/Mentor/Qualifier/Evaluator memo, formal training completed and recommended training completed.
- (U) Tab 7- (As Required) Training activity, changes, lapses in training, progress, comments.
- (U) Tab 8- (Mandatory) Signature page.

(U) Proficiency Level | Work Function

(U) This JQR contains tasks performed by this work role at the Basic proficiency level. The KSAs in Tab 2 indicate the level of proficiency required for different levels of performance.

(U) Work Description

(U) This JQR describes the required tasks performed in **Cyberspace Capability Developer** work role at the Basic proficiency level and its associated joint tasks and KSAs. This JQR includes the joint tasks and KSAs required of this work role, as well as position/team/service-specific requirements that are not part of the joint standard.

(U) Minimum Qualification Requirements

(U) To meet the minimum qualifications as an entering Basic **Cyberspace Capability Developer**, the Member¹ must have completed all required Training and Certification requirements listed in Tab 3. Further, it is expected that the Member will maintain all required certifications and complete any additional training, as specified by the Member's leadership, while in the position.

¹ The term "Member" is used throughout as the individual for whom the JQR is written; it is not limited to only members of the CMF

(U) Tab 1. About the Work: Cyberspace Capability Developer

(U) Mission

(U) As part of the Cyber Mission Forces (CMF), the Cyberspace Capability Developer provides support to CMF teams via the development or modification of customized tools or capabilities. The supported CMF teams include: National Mission Teams (NMT), National Support Teams (NST), Combatant Mission Teams (CMT), Combat Support Teams (CST), and Cyber Protection Teams (CPT).

(U) Proficiency Levels

(U) Proficiency levels define how an individual must perform the qualification task at a given proficiency level of Basic. Proficiency in qualification in a task is cumulative, meaning that those at the Master level have met proficiency requirements for Basic and Senior tasks.

Knowledge Levels	
A	Can identify basic facts and terms about a subject
B	Can identify relationships of basic facts and state general principles about the subject
C	Can analyze facts and principles and draw conclusions about the subject.
D	Can evaluate conditions and make proper decisions about the subject.
Skill/Ability Levels	
1	Must be familiar with this competency and be generally capable of independently handling simple tasks or assignments.
2	Must be capable of independently handling some complex tasks or assignments related to this competency but may need direction and guidance on others.
3	Must be capable of independently handling a wide variety of complex and/or high profile tasks or assignments related to this competency. Must be an authority in this area and/or often sought out by others for advice or to teach/mentor others on highly complex or challenging tasks or assignments related to this competency.

Joint Cyberspace Training and Certification Standards (JCT&CS) Proficiency Levels	
Basic	Basic/Developmental/Apprentice
Senior	Full Performance/Journeyman
Master	Advanced/Expert/Master

(U) Tab 2. Basic Cyberspace Capability Developer Tasks and KSAs created in a Cyber Standards Analysis Team (CSAT) 19-21 April 2022 and to be codified in the Joint Cyber Training and Certification Standards (JCT&CS)

(U) Tasks

DCWF #	USCYBERCOM #	Task	Basic Proficiency Level
414		(U) Analyze user needs and software requirements to determine feasibility of design within time and cost constraints.	Basic
515A		(U) Develop software system testing and validation procedures, programming, and documentation.	Basic
543		(U) Develop secure code and error handling.	Basic
630		(U) Identify and direct the remediation of technical problems encountered during testing and implementation of new systems (e.g., identify and find work-around for communication protocols that are not interoperable).	Basic
709A		(U) Modify and maintain existing software to correct errors, to adapt it to new hardware, or to upgrade interfaces and improve performance.	Basic
756		(U) Perform integrated quality assurance testing for security functionality and resiliency attack.	Basic
764		(U) Perform secure programming and identify potential flaws in codes to mitigate vulnerabilities.	Basic
785		(U) Prepare detailed workflow charts and diagrams that describe input, output, and logical operation, and convert them into a series of instructions coded in a computer language.	Basic
1149A		(U) Enable applications with public keying by leveraging existing public key infrastructure (PKI) libraries and incorporating certificate management and encryption functionalities when appropriate.	Basic
1151		(U) Identify and leverage the enterprise-wide version control system while designing and developing secure applications.	Basic
6780		(U) Utilize different programming languages to write code, open files, read files, and write output to different files.	Basic

UNCLASSIFIED

DCWF #	USCYBERCOM #	Task	Basic Proficiency Level
	JT0059	(U) Create or enhance cyberspace capabilities to compromise, deny, degrade, disrupt, destroy, or manipulate automated information systems.	Basic
	JT0060	(U) Create or enhance cyberspace solutions to enable surveillance and reconnaissance of automated information systems.	Basic
	JT0061	(U) Reference capability repositories and other sources to identify existing capabilities which fully/partially meet customer requirements (with or without modification).	Basic
	JT0062	(U) Analyze, modify, develop, debug, and document software and applications which run in user space.	Basic
	JT0063	(U) Analyze, modify, develop, debug, and document software and applications utilizing standard, non-standard, specialized, and/or unique communication protocols.	Basic
	JT0066	(U) Develop, modify, and utilize automation technologies to enable employment of capabilities as efficiently as possible (e.g. TDD, CI/CD, etc.)	Basic
	JT0067	(U) Analyze and document applications using assembly languages.	Basic
	JT0068	(U) Utilize tools to decompile, disassemble, analyze, and reverse engineer compiled binaries.	Basic
	JT0070	(U) Perform static and dynamic analysis in order to find errors and flaws.	Basic
	JT0072	(U) Design and develop data storage requirements, database structure, process flow, systematic procedures, algorithms, data analysis, and file structures.	Basic
	JT0073	(U) Utilize data structures to organize, sort, and manipulate elements of information.	Basic
	JT0074	(U) Design and develop user interfaces (e.g. web pages, GUIs, CLIs, Console Interfaces).	Basic
	JT0075	(U) Utilize secure coding techniques during development of software and applications.	Basic
	JT0076	(U) Apply cryptography primitives to protect the confidentiality and integrity of sensitive data.	Basic
	JT0078	(U) Produce artifacts to inform risk analysis, acceptance testing, and legal review.	Basic

DCWF #	USCYBERCOM #	Task	Basic Proficiency Level
	JT0079	(U) Locate and utilize technical specifications and industry standards (e.g. Internet Engineering Task Force (IETF), IEEE, IEC, and International Standards Organization (ISO)).	Basic
	JT0081	(U) Apply software engineering best practices to enable sustainability and extensibility (Agile, TDD, CI/CD, etc.) to include containerization and virtualization technologies.	Basic
	JT0082	(U) Enter work into task and project management tools used for software development (e.g. Jira, Confluence, Trac, MediaWiki, etc.).	Basic
	JT0224	(U) Develop content for cyber capabilities.	Basic
	JT0225	(U) Generate proper supporting documentation of cyber capability.	Basic
	JT0227	(U) Analyze countermeasures and mitigations against potential exploitations of programming language weaknesses and vulnerabilities in system and elements.	Basic

(U) Knowledge Skills and Abilities (KSA)

DCWF #	USCYBERCOM #	KSA	Basic Proficiency Level
22		(U) Knowledge of computer networking concepts and protocols, and network security methodologies.	B
23		(U) Knowledge of computer programming principles such as object-oriented design.	B
27		(U) Knowledge of cryptography and cryptographic management concepts.	B
40		(U) Knowledge of organization's evaluation and validation requirements.	A
56		(U) Knowledge of cybersecurity principles and methods that apply to software development.	B
63		(U) Knowledge of cybersecurity principles and organizational requirements (relevant to confidentiality, integrity, availability, authentication, nonrepudiation).	B

UNCLASSIFIED

DCWF #	USCYBERCOM #	KSA	Basic Proficiency Level
74		(U) Knowledge of low-level computer languages (e.g., assembly languages).	B
95A		(U) Knowledge of penetration testing principles, tools, and techniques.	A
102		(U) Knowledge of programming language structures and logic.	C
116		(U) Knowledge of software debugging principles.	C
118		(U) Knowledge of software development models (e.g., Waterfall Model, Spiral Model).	A
119		(U) Knowledge of software engineering.	B
278		(U) Knowledge of different types of network communication (e.g., LAN, WAN, MAN, WLAN, WWAN).	B
320A		(U) Knowledge of external organizations and academic institutions with cyber focus (e.g., cyber curriculum/training and Research and Development).	A
904		(U) Knowledge of interpreted and compiled computer languages.	B
905		(U) Knowledge of secure coding techniques.	B
979		(U) Knowledge of supply chain risk management standards, processes, and practices.	A
1036		(U) Knowledge of applicable laws (e.g., Electronic Communications Privacy Act, Foreign Intelligence Surveillance Act, Protect America Act, search and seizure laws, civil liberties and privacy laws), statutes (e.g., in Titles 10, 18, 32, 50 in U.S. Code), Presidential Directives, executive branch guidelines, and/or administrative/criminal legal guidelines and procedures relevant to work performed.	A
1056		(U) Knowledge of operations security.	A
1062		(U) Knowledge of software reverse engineering techniques	B
1159		(U) Knowledge of cyber threats and vulnerabilities.	B
3140		(U) Knowledge of basic programming concepts (e.g., levels, structures, compiled vs. interpreted languages).	C
3146		(U) Knowledge of both internal and external customers and partner organizations, including information needs, objectives, structure, capabilities, etc.	A

DCWF #	USCYBERCOM #	KSA	Basic Proficiency Level
3441		(U) Knowledge of physical and logical network infrastructure, to include hubs, switches, routers, firewalls, etc.	B
3622		(U) Knowledge of organizational and partner authorities, responsibilities, and contributions to achieving objectives.	A
	JK0102	(U) Knowledge of cyber mission force equipment taxonomy (Platform-Access-Payloads/Toolset), capability development process and repository.	A
	JK0103	(U) Knowledge of cyber adversary threat tier taxonomy (2014 National Intelligence Estimate [NIE]), DIA/NSA Standard Cyber Threat Model, etc.).	A
	JK0104	(U) Knowledge of sources and locations of cyber capability registries and repositories (e.g. Joint Cyber Tactics Manual (JCTM), Cyber Capability Registry (CCR), Agency and service repositories, etc.).	A
	JK0105	(U) Knowledge of sources and locations (public and classified) of capability development TTPs and tradecraft information/intelligence used by the US Gov and others.	A
	JK0107	(U) Knowledge of the supported organization's approval process for operational use of a capability.	A
	JK0108	(U) Knowledge of relevant mission processes including version control processes, release processes, documentation requirements, and testing requirements.	B
	JK0109	(U) Knowledge of modern software development methodologies (e.g. Continuous Integration (CI), Continuous Delivery (CD), Test Driven Development (TDD), etc.).	B
	JK0110	(U) Knowledge of your organization's project management, timeline estimation, and software engineering philosophy (e.g. CI/CD, TDD, etc.).	B
	JK0111	(U) Knowledge of principles, methodologies, and tools used to improve quality of software (e.g. regression testing, test coverage, code review, pair programming, etc.).	A

UNCLASSIFIED

DCWF #	USCYBERCOM #	KSA	Basic Proficiency Level
	JK0112	(U) Knowledge of terms and concepts of operating system fundamentals (e.g. virtualization, paging, file systems, I/O, memory management, process abstraction, etc.).	B
	JK0114	(U) Knowledge of the use and application of static and dynamic program analysis.	B
	JK0116	(U) Knowledge of data serialization formats (e.g. XML, JSON, etc.).	B
	JK0117	(U) Knowledge of the concepts and terminology of data structures and associated algorithms (e.g., search, sort, traverse, insert, delete).	C
	JK0118	(U) Knowledge of task and project management tools used for software development (e.g. Jira, Confluence, Trac, MediaWiki, etc.).	B
	JK0238	(U) Knowledge of embedded systems.	A
	JK0298	(U) Knowledge of techniques to harden capabilities to prevent attacks and forensics.	A
168		(U) Skill in conducting software debugging.	2
185A		(U) Skill in developing applications that can log and handle errors, exceptions, and application faults and logging.	2
973A		(U) Skill in using code analysis tools.	1
1020A		(U) Skill in secure test plan design (e.g. unit, integration, system, and acceptance).	1
1140A		(U) Skill in using Public-Key Infrastructure (PKI) encryption and digital signature capabilities into applications (e.g., S/MIME email, SSL traffic).	2
	JS0129	(U) Skill in conducting "open source" research.	2
1071A		(U) Ability to develop secure software according to secure software deployment methodologies, tools, and practices.	2
3022		(U) Ability to communicate complex information, concepts, or ideas in a confident and well-organized manner through verbal, written, and/or visual means.	1
	JA0232	(U) Ability to analyze, modify, develop, debug, and document software and applications in C programming language.	2
	JA0233	(U) Ability to analyze, modify, develop, debug, and document software and applications in Python programming language.	2

UNCLASSIFIED

UNCLASSIFIED

DCWF #	USCYBERCOM #	KSA	Basic Proficiency Level
	JA0234	(U) Ability to analyze, modify, develop, debug, and document software and applications utilizing standard, nonstandard, specialized, serialization and/or unique network communication protocols.	2
	JA0235	(U) Ability to interpret customer requirements and evaluate resource and system constraints to create solution design specifications.	2
	JA0236	(U) Ability to use reference documentation for C, Python, assembly, and other international technical standards and specifications (IEEE, ISO, IETF, etc.).	2
	JA0238	(U) Ability to use common networking protocols.	2
	JA0239	(U) Ability to use data structures.	2

(U) Tab 3. (Mandatory) Basic Cyberspace Capability Developer Formal Training Requirements as outlined on the USCYBERCOM CMF Training Pipeline

(U) Since USCYBERCOM does not have a validated training solution for capability developer, Services should utilize Service-level capability developer training or qualification pathways.

(U) **Air Force:**

- (U) Officers: Undergraduate Cyber Warfare Training (UCWT)
- (U) Enlisted: Tech School and Initial Developer Fundamentals (IDF)
- (U) Civilians: Initial Developer Fundamentals (IDF)
- (U) **Note:** Individuals directly reporting to the unit may test out via CCD Qualification Exam in lieu of formal training

(U) **Army:**

- (U) Service-level capability developer training and/or qualification pathways

(U) **Navy:**

- (U) Officers/Enlisted/Civilians: CMF-CCD Basic JQR Courses and Advanced Cyber Training Program (ACTP) CNO Programming capstone

(U) Tab 4. (Mandatory) Basic Level Cyberspace Capability Developer JQS line items linked to approved Tasks and KSAs

(U) JOINT KNOWLEDGE AND SKILLS

(U) The Joint Standard for this work role requires that all Members have core foundational knowledge and skills. These core KSAs are required for anyone performing this work role, regardless of position, team, mission, or other condition.

(U) Using Work Role Task and KSA Value levels (A, B, C, D) provided in the CYBERSPACE TECHNICAL MANUAL (CTM) 7-0.1, JOINT CYBERSPACE TRAINING AND CERTIFICATION STANDARDS (JCT&CS) VERSION 4.0, dated 18 SEPTEMBER 2020, as outlined in Tab 2, indicate the level of knowledge of the incoming Member, Initial and date to indicate the Member has the requisite Core Knowledge of each item listed. If the Member has less than level “A” knowledge, indicate this with a zero (“0”).

(U) The following statements capture the core knowledge and skills information required by all members in the Basic Cyberspace Capability Developer. This section verified the member’s knowledge level in this core set of KSAs.

Tab 4.1. Basic Cyberspace Capability Developer (CCD)

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
Module 1 - (U) Cyberspace Operations Fundamentals					
Training Resources & Technical References: <ul style="list-style-type: none"> JP 3-12 					
1.1	(U) Describe the mission of the following Cyber Mission Force (CMF) Elements: <ol style="list-style-type: none"> Cyber National Mission Force (CNMF) Cyber Combat Mission Force (CCMF) Cyber Protection Force (CPF) 	DCWF1151 DCWF3146 DCWF3622 JK0102			
1.2	(U) Describe the mission of the following CMF teams: <ol style="list-style-type: none"> Combat Mission Team (CMT) National Mission Team (NMT) Combat Support Team (CST) National Support Team (NST) 	DCWF3146 DCWF3622			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	e. Cyber Protection Team (CPT)				
1.3	(U) Read and understand the following cyberspace operations doctrine: a. JP 3-12 (Cyberspace Operations) b. USCYBERCOM Operational Guidance 3-2 (DCO) c. USCYBERCOM Cyber Warfare Publication (CWP 3-33.4)	DCWF1036 DCWF1151 DCWF3146 DCWF3622 JK0108			
1.4	(U) Describe the mission and responsibilities of key organizations: a. U.S. Cyber Command (USCC) b. Director Operation Test and Evaluation (DOTE) c. Federally Funded Research and Development Centers (FFRDC) d. Central Intelligence Agency (CIA) e. Federal Bureau of Investigation (FBI) f. Department of Homeland Security (DHS) g. Defense Threat Reduction Agency (DTRA) h. Defense Digital Service (DDS) i. National Security Agency (NSA) j. National Air and Space Intelligence Center (NASIC)	DCWF320 A DCWF3146 DCWF3622			
1.5	(U) Demonstrate knowledge of U.S. Code and its application to the Intelligence Community (IC) and Cyberspace Operations (Title 10, 18, and 50).	DCWF1036			
1.6	(U) Describe the classification levels of the development networks and the associated handling requirements of source code and binaries for your organization.	DCWF1056 JK0105			
1.7	(U) Demonstrate knowledge of sources and locations (public and classified) of capability development TTPs and tradecraft information/intelligence used by the US Gov and others.	DCWF1159 JT0061 JK0104 JK0105			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
Module 2 - (U) DCO Fundamentals Training Resources & Technical References: <ul style="list-style-type: none"> • CWP 3-33.4 • JP 3-12 					
2.1	(U) Describe the overall mission of a Cyber Protection Team (CPT).	DCWF3146 DCWF3622			
2.2	(U) Describe the CPT Support Element and Mission Element and their capabilities.	DCWF3146			
2.3	(U) Describe CPT mission types: <ul style="list-style-type: none"> a. National CPT b. DoDIN CPT c. Combatant Command CPT d. Service CPT 	DCWF3146			
2.4	(U) Describe the following DCO work roles: <ul style="list-style-type: none"> a. Cyber Operations Planner b. Network Analyst c. Host Analyst d. Analytic Support Officer e. Data Engineer f. Network Technician g. All-Source Analyst 	DCWF3146			
2.5	(U) Read and understand the DNI threat assessments (e.g. National Intelligence Estimate (NIE), DIA/NSA Standard Cyber Threat Model, etc.).	DCWF979 JT0078 JK0103			
Module 3 - (U) OCO Fundamentals					
3.1	(U) Describe the difference between conducting Cyberspace Intelligence, Surveillance, and Reconnaissance (C-ISR) and Cyberspace Surveillance and Reconnaissance (C-SR).	DCWF1036 DCWF3622 JK0107			
3.2	(U) Describe the following OCO work roles: <ul style="list-style-type: none"> a. Remote Operator (ION, RO) b. Mission Commander (MC) 	DCWF3146			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	c. Exploitation Analyst (EA) d. Digital Network Exploitation Analyst (DNEA) and Target Digital Network Analyst (TDNA) e. Operational Target Development Analyst (OTDA), Targeteer, and Fire Support Planner f. Cyber Operations Planner g. Team Lead and Deputy Team Lead h. Language Analyst (LA) i. Target Analyst Reporter (TAR)				
Module 4 - (U) Mission Process					
4.1	(U) Describe the CCDO standard of interaction between a capability development organization and its higher requirements-generating headquarters, including pre-requirement planning, drafting requirements, and interacting with tool champions, product owners, or requirement-owning stakeholders.	DCWF40 DCWF414 DCWF515 A DCWF3146 DCWF3622 JT0061 JT0078 JK0102 JK0107 JK0110 JA0235			
4.2	(U) Describe the U.S. Cyber Command testing and evaluation process, including Developmental Test and Evaluation, Developmental Acceptance, Evaluated Level of Assurance, and Operational Test and Evaluation. Training Resources & Technical References: <ul style="list-style-type: none"> USCCI 3801-19 	DCWF40 DCWF63 DCWF515 A JT0076 JK0107 JK0108			
4.3	(U) Describe an example of software development/documentation best practices.	DCWF56 DCWF785 DCWF764 DCWF756			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
		DCWF543 DCWF515 A DCWF119 DCWF905 DCWF1151 DCWF116 DCWF118 JT0082 JT0075 JT0070 JT0066 JK0111 JK0118			
Module 5 - (U) Style Fundamentals					
5.1	(U) With references, resources, and a provided unit-level coding standard or style guide, identify coding standard violations in Python source code.	DCWF973 A JT0070 JT0079 JK0111 JA0233 JA0236			
5.2	(U) With references, resources, and a provided unit-level coding standard or style guide, identify coding standard violations in C source code.	DCWF973 A JT0070 JT0079 JK0111 JA0232 JA0236			
Module 6 - (U) C Programming Training Resources & Technical References: <ul style="list-style-type: none"> • The C Programming Language (Kernighan, Ritchie) • Unix man pages • C Primer Plus (Prata) 					

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
6.1	(U) Describe the purpose and use of C programming fundamentals: <ol style="list-style-type: none"> The main() function The return statement Macro guards Data types Functions and procedures Parameters Scope Return values (return type and reference) Header files Keywords (static and extern) Pointers An array C preprocessor Casting Control flow Endianness Multi-byte vs. Unicode character sets Multi-threading Hashing 	DCWF904 DCWF3140 JA0232 JA0236			
6.2	(U) Describe C programming concepts in regards to memory: <ol style="list-style-type: none"> Memory map of a Linux process Automatically allocated memory Dynamically allocated memory Statically allocated memory In the context of automatic vs dynamic allocation, explain how those concepts are related to the implementation of a stack and heap in a C program 	DCWF3140			
6.3	(U) Demonstrate the proper declaration, understanding, and use of C data types and underlying structures: <ol style="list-style-type: none"> char short 	DCWF3140 JA0232			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	<ul style="list-style-type: none"> c. int d. long e. long long f. float g. double h. long double 				
6.4	(U) Demonstrate proper declaration, understanding, and use of fixed-width C data types defined in stdint.h: <ul style="list-style-type: none"> a. int8_t b. uint8_t c. int16_t d. uint16_t e. int32_t f. uint32_t g. int64_t h. uint64_t 	DCWF3140 JA0232			
6.5	(U) Demonstrate the ability to create and implement a function that uses different arrays: <ul style="list-style-type: none"> a. An array b. A multi-dimensional array 	DCWF3140 JA0232			
6.6	(U) Demonstrate the ability to perform basic arithmetic operations using appropriate C operators while ensuring proper order of operations (PEMDAS): <ul style="list-style-type: none"> a. Addition b. Subtraction c. Multiplication d. Division e. Modulus (%) f. Pre-Increment (++i) g. Post-Increment (i++) h. Pre-Decrement (--i) i. Post-Decrement (i--) 	DCWF3140 JA0232			
6.7	(U) Demonstrate the ability to properly use the	DCWF3140			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	standard main() entry arguments: a. int argc b. char *argv[]	JA0232			
6.8	(U) Demonstrate the ability to perform file management operations in C: a. Open an existing file b. Read data from a file c. Write data to a file d. Modify data in a file e. Close an open file f. Print file information to the console g. Create a new file h. Append data to an existing file i. Delete a file j. Determine the size of a file (in a UNIX-based operating system) k. Determine location within a file l. Insert data into an existing file	DCWF3140 DCWF6780 JA0232			
6.9	(U) Demonstrate the ability to create and implement functions to meet a requirement: a. Proper declaration for created functions b. A function that does not return a value (i.e., is declared void) c. A function that is passed an argument by value d. A function that takes a pointer argument e. A function that returns a value using a return statement f. A function that modifies an output parameter through a pointer g. A function that receives input from a user h. A function pointer i. A recursive function	DCWF3140 JA0232			
6.10	(U) Demonstrate the ability to perform data validation: a. Validating input received matches input	DCWF3140 JA0232			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	expected				
6.11	(U) Demonstrate skill in using pointers: <ol style="list-style-type: none"> Declaring an integer pointer Dereferencing a variable to get its value Printing the address of the variable Assigning a value to a pointer Make use of a function pointer to call another function Make effective use of pointer arithmetic to traverse an array 	DCWF3140 JA0232			
6.12	(U) Demonstrate skill in creating and implementing conditional statements, expressions, and constructs: <ol style="list-style-type: none"> for loop while loop do while loop if statement if/else statement if/else if/else statement switch statement effective use of goto labels to construct a single exit point within a function 	DCWF3140 JA0232			
6.13	(U) Demonstrate skill in creating and implementing a sort routine.	DCWF3140 JA0232			
6.14	(U) Given a specification for a stateful application or protocol, describe and/or draw a diagram of the possible states it can have. <ol style="list-style-type: none"> Show or describe criteria necessary to change between each possible state Show or describe the input/output generated (if any) during the change between each possible state Show or describe the conditions necessary to be in the initial state Show or describe the conditions necessary to get to the final state 	JA0234			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	<ul style="list-style-type: none"> e. Show or describe the relationship (if any) between each state in the program f. Account for any possible error conditions that may occur during state transitions 				
6.15	(U) Describe terms associated with compiling, linking, debugging, and executables: <ul style="list-style-type: none"> a. Portable Executable (PE) b. Executable and Linkable Format (ELF) c. Difference between PE and ELF d. Difference between a library (shared object / DLL) and a regular executable program e. Calling convention/Application Binary Interface (ABI) 	DCWF168 DCWF3140 JA0232			
6.16	(U) Demonstrate skill in compiling, linking, and debugging: <ul style="list-style-type: none"> a. Execute a program in a debugger to perform general debugging actions b. Create a program using the compilation and linking process c. Compile position-independent code using a cross-compiler 	DCWF116 DCWF168 DCWF630 DCWF709 A DCWF3140 JA0232			
6.17	(U) Demonstrate the ability to build a binary from multiple C source files and headers by writing a Makefile using explicit rules.	DCWF3140 JA0232			
6.18	(U) Describe how and when bitwise operators are used: <ul style="list-style-type: none"> a. and (&) b. or () c. xor (^) d. bitwise complement (~) e. shift left (<<) f. shift right (>>) g. Add, removing, and testing for single-bit flags h. Extracting arbitrary bytes from multi-byte 	DCWF3140 JA0232			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	data types				
6.19	(U) Demonstrate skill in using the C preprocessor.	DCWF3140 JA0232			
6.20	(U) Demonstrate skill in accessing environment variables.	DCWF3140 JA0232			
6.21	(U) Demonstrate skill in controlling memory: <ol style="list-style-type: none"> With attention given to implementation defined behavior, compare and contrast standard memory allocation functions (e.g., malloc(), calloc(), realloc(), and free()) Demonstrate appropriate error checking when managing memory allocations Describe programming techniques that reduce the occurrence of memory leaks (e.g., behaviors that reinforce a clear ownership model) Demonstrate effective use of Valgrind with --leak-check=full to identify memory leaks Given code samples, identify and remove memory leaks 	DCWF973 A DCWF3140 JA0232			
Module 7 - (U) Python Programming					
Training Resources & Technical References: <ul style="list-style-type: none"> Python 3 online documentation 					
7.1	(U) Describe purpose and use of foundational Python mechanics: <ol style="list-style-type: none"> The return statement Data types A function Parameters Scope Return values (return type and reference) Import files Dictionaries 	DCWF3140 JA0236 JA0233			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	<ul style="list-style-type: none"> i. List j. Tuple k. Singleton l. The term mutable m. The term immutable 				
7.2	(U) Demonstrate the proper declaration and use of Python data types and object-oriented constructs: <ul style="list-style-type: none"> a. Integer (int) b. Float (float) c. String (str) d. List (list) e. Multi-dimensional list f. Dictionary (dict) g. Tuple (tuple) h. Singleton 	DCWF3140 JA0233			
7.3	(U) Demonstrate the ability to perform basic arithmetic operations using Python operators while ensuring proper order of operations (PEMDAS): <ul style="list-style-type: none"> a. Addition b. Subtraction c. Multiplication d. Division e. Modulus 	DCWF3140 JA0233			
7.4	(U) Demonstrate the ability to perform file management operations in Python: <ul style="list-style-type: none"> a. Open an existing file b. Read data from a file c. Parse data from a file d. Write data to a file e. Modify data in a file f. Close an open file g. Print file information to the console h. Create a new file i. Append data to an existing file j. Delete a file 	DCWF3140 DCWF6780 JA0233			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	k. Determine the size of a file l. Determine location within a file m. Insert data into an existing file				
7.5	(U) Demonstrate the ability to create and implement functions to meet a requirement: a. A function that returns multiple values b. A function that receives input from a user c. A recursive function	DCWF3140 JA0233			
7.6	(U) Demonstrate the ability to perform data validation: a. Validating received input matches expected input b. Designing and implementing a scheme for exception handling	DCWF185 A DCWF3140 JA0233			
7.7	(U) Demonstrate skill in creating and implementing conditional statements, expressions, and constructs: a. for loop b. while loop c. with statement d. if statement e. if/else statement f. if/elif/else statement g. try/except/finally	DCWF3140 JA0233			
7.8	(U) Describe the terms and fundamentals associated with object oriented programming using Python: a. Class b. Object c. Difference between an object when discussing a class d. Advantages to object-oriented programming e. Inheritance f. The keyword "super" g. Initialization function of a constructor	DCWF23 DCWF119 DCWF3140 DCWF6780 JA0233			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	h. The keyword "self" i. The getter and setter functions j. Attributes of a class k. Factory design pattern l. Singleton design pattern m. Adapter design pattern n. Bridge design pattern Training Resources & Technical References: <ul style="list-style-type: none"> Design Patterns: Elements of Reusable Object-Oriented Software (Gamma, Helm, Johnson, Vlissides) 				
7.9	(U) Demonstrate the ability to parse command line arguments using built-in functionality.	DCWF3140 JA0233			
Module 8 - (U) Data Structures Training Resources & Technical References: <ul style="list-style-type: none"> Data Structures and Algorithms Made Easy: Data Structures and Algorithmic Puzzles 5th edition (Narasimah Karumanchi) 					
8.1	(U) Describe the concepts and terms associated with key data structures: <ol style="list-style-type: none"> Hash table Stack Tree vs Binary search tree Linked list Double linked list Queue vs Priority Queue Circularly linked list Weighted graph Common pitfalls when using linked lists, trees, and graphs The effect of First In First Out (FIFO) and Last In First Out (LIFO) 	DCWF102 JT0073 JK0117 JA0239			
8.2	(U) Demonstrate skill in creating and using a circularly linked list that accepts any data type:	DCWF102 JT0073			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	<ul style="list-style-type: none"> a. Creating a circularly linked list with n number of items b. Navigating through a circularly linked list c. Finding the first occurrence of an item in a circularly linked list d. Sorting the circularly linked list alphanumerically using a function pointer e. Removing selected items from the circularly linked list f. Inserting an item into a specific location in a circularly linked list g. Removing all items from the circularly linked list h. Destroying a circularly linked list 	JK0117 JA0239			
8.3	(U) Demonstrate skill in creating and using a binary search tree that accepts any data type: <ul style="list-style-type: none"> a. Creating a binary search tree with n number of items b. Navigating through a binary search tree c. Locating an item in a binary search tree d. Removing selected items from the binary search tree e. Removing all items from the binary search tree f. Describe implementation strategies for a balanced binary search tree g. Destroying a binary search tree 	DCWF102 JT0073 JK0117 JA0239			
8.4	(U) Demonstrate skill in creating and using a hash table that accepts any data type: <ul style="list-style-type: none"> a. Creating a hash table with n number of items b. Navigating through a hash table to find the nth item c. Finding an item in a hash table d. Removing selected items from a hash table e. Inserting an item into a hash table 	DCWF102 JT0073 JK0117 JA0239			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	f. Implement functionality to mitigate hash collisions within the hash table g. Removing all items from the hash table				
8.5	(U) Demonstrate skill in creating and using a stack that accepts any data type: a. Create a stack (cannot be fixed sized) b. Adding an item in a stack (enforce FILO) c. Removing n items from a stack d. Removing all items from the stack e. Destroying a stack f. Preventing a stack overrun	DCWF102 JT0073 JK0117 JA0239			
8.6	(U) Demonstrate skill in implementing a priority queue that accepts any data type: a. Defining the underlying structures required for priority queues (cannot be fixed sized) b. Assigning a priority to each element c. Inserting an element into the priority queue d. Removing the element with the highest priority from the priority queue e. Destroying a priority queue f. Define possible applications of a priority queue	DCWF102 JT0073 JK0117 JA0239			
Module 9 - (U) Algorithms					
9.1	(U) Describe concepts associated with traversal techniques: a. Depth first traversal b. Breadth first traversal c. The technique of determining the weight of a given path when traversing a graph d. How the most efficient path for traversing a graph is determined	DCWF102 DCWF119 JT0072 JK0117			
9.2	(U) Describe concepts associated with hashing: a. Data distribution as it relates to hashing	DCWF102 DCWF119 JT0072			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	b. Hash function efficiency c. Hash collisions				
9.3	(U) Demonstrate the ability to analyze sorting routines to determine the most efficient one to use, using an approximation of Big-O notation. a. Insertion sort b. Selection sort c. Merge sort d. Heap sort e. Quick sort f. Hashing	DCWF102 DCWF119 JT0072 JT0073 JK0117 JA0239			
Module 10 - (U) Operating System Concepts Training Resources & Technical References: <ul style="list-style-type: none"> Operating Systems: Three Easy Pieces (Arpaci-Dusseau) 					
10.1	(U) Describe terms and concepts associated with Operating System (OS) virtualization: a. Processes b. CPU scheduling c. Paging tables d. Caching e. Kernel and user-mode memory	JK0112			
10.2	(U) Describe the following terms and concepts: a. File systems b. The boot process	JK0112			
10.3	(U) Demonstrate the ability to use the following constructs: a. Interrupts b. Signal handling	JK0112			
10.4	(U) Describe terms and concepts associated with concurrency: a. Threading (thread vs pthread) b. fork c. join d. create	JK0112			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	e. exit f. detach g. self h. Locking (mutex, semaphore, etc) i. Race conditions j. Deadlocks k. thread safe l. thread id m. conditional variables n. atomics o. Scheduling algorithms, i.e., round robin, shortest job first, priority scheduling, etc.				
10.5	(U) Demonstrate the ability to use the following constructs associated with concurrency: a. Threads b. Locks c. Condition variables d. Atomics e. Thread Pool (with graceful shutdown without memory leaks)	JK0112			
Module 11 - (U) Secure Coding Training Resources & Technical References: <ul style="list-style-type: none"> Secure Coding in C and C++ 2nd edition (Robert Seacord) 					
11.1	(U) Describe terms and concepts associated with secure coding practices: a. Common string-handling functions b. Which functions guarantee null terminated strings c. An off-by-one error d. An integer overflow e. A buffer overflow f. The concept of use-after-free g. Resource acquisition is initialization (RAII)	DCWF27 DCWF56 DCWF95A DCWF905 DCWF1071 A JT0076 JT0227 JK0298			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	h. The difference between input validation vs. input sanitization i. The meaning of a pure function and if a function has a side-effect j. General low-level crypto basics (e.g. different encryption schemes and how you might implement them system wide, what crypto is better for different use cases) k. Penetration testing principles, tools, and techniques l. Obfuscation techniques				
11.2	(U) Demonstrate skill in using secure coding techniques: a. Formatting string vulnerabilities b. Safe buffer size allocation c. Input sanitization d. Input validation e. Establish a secure communications channel using an SSL library f. Securely zeroing-out memory (compiler optimizations)	DCWF543 DCWF764 DCWF905 DCWF1071 A DCWF1140 A DCWF1149 A JT0075 JT0076			
Module 12 - (U) Networking Fundamentals Training Resources & Technical References: <ul style="list-style-type: none"> • Beej's guide to internet programming using internet sockets (Jorgensen) • Computer Networking: A Top-Down Approach 7th edition (James Kurose) 					
12.1	(U) Describe the concepts and terms associated with networking fundamentals: a. Transmission Control Protocol (TCP) / User Datagram Protocol (UDP) b. Open Systems Interconnect (OSI) model c. POSIX API/BSD sockets d. Purpose and use of sockets e. Request For Comments (RFCs)	DCWF22			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	f. Purpose of subnetting				
12.2	(U) Describe the concepts and terms associated with common protocols and their associated ports, if applicable: a. Address Resolution Protocol (ARP) b. Hypertext Transfer Protocol/Secure (HTTP/HTTPS) c. Domain Name System (DNS) d. Simple Mail Transfer Protocol (SMTP) e. Internet Control Message Protocol (ICMP) f. Dynamic Host Configuration Protocol (DHCP) g. Internet Protocol version 4 (IPv4) h. Internet Protocol version 6 (IPv6)	DCWF22			
12.3	(U) Describe the addressing associated with key networking protocols: a. IPv4 b. IPv6 c. Ethernet	DCWF22			
12.4	(U) Describe the concepts and terms associated with physical and logical network infrastructure: a. Hubs b. Switches c. Routers d. Firewalls	DCWF22 DCWF3441			
12.5	(U) Describe different types of network communications: a. LAN b. WAN c. MAN d. WLAN e. WWAN	DCWF22 DCWF278			
12.6	(U) In Python, demonstrate skill in using networking commands accounting for endianness: a. socket()	DCWF22 JA0238			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	b. send() c. recv() d. sendto() e. recvfrom() f. bind() g. listen() h. connect() i. accept() j. close() k. gethostname()				
12.7	(U) In C, demonstrate skill in using networking commands accounting for endianness: a. socket() b. send() c. recv() d. sendto() e. recvfrom() f. bind() g. connect() h. accept() i. getsockopt() j. setsockopt() k. getaddrinfo() l. gethostname() m. struct sockaddr n. struct sockaddr_in o. struct sockaddr_un p. struct sockaddr_storage	DCWF22			
12.8	(U) Demonstrate skill in handling partial send()/recv().	DCWF22			
12.9	(U) Demonstrate skill in implementing functions that can properly handle any IP address (IPv4/IPv6).	DCWF22			
12.10	(U) Demonstrate skill in IO Multiplexing. a. select	DCWF22			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	b. poll				
Module 13 - (U) Serialization					
13.1	(U) Describe data serialization and deserialization.	JT0072 JK0116			
13.2	(U) Demonstrate the ability to handle partial reads and writes during serialization and deserialization.	DCWF785 JT0063 JT0072 JK0116 JA0234			
13.3	(U) Demonstrate the ability to serialize fixed size multi-byte types between systems of differing endianness.	DCWF785 JT0063 JT0072 JK0116 JA0234			
13.4	(U) Demonstrate the ability to serialize and deserialize variable sized data structures between systems of differing endianness.	DCWF785 JT0063 JT0072 JK0116 JA0234			
13.5	(U) Describe libraries commonly used to aid in serialization.	JT0072 JK0116			
Module 14 - (U) Regular Expressions					
14.1	(U) Describe the following concepts related to regular expressions: a. Basic Regular Expressions (BRE) b. Extended Regular Expressions (ERE) c. Perl Compatible Regular Expressions	DCWF3140			
14.2	(U) In C, demonstrate the ability to incorporate regular expression processing into a program a. Matching b. Use of capture groups	JA0232			
14.3	(U) In Python, demonstrate the ability to incorporate regular expression processing into a program a. Matching	JA0233			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	b. Use of capture groups				
Module 15 - (U) Reverse Engineering					
Training Resources & Technical References: <ul style="list-style-type: none"> Practical Reverse Engineering (Dang, Gazet, Bachaalany) 					
15.1	(U) Describe the purpose of the following registers: a. General Purpose Registers b. EIP c. EFLAGS	DCWF74 JT0067			
15.2	(U) Describe the difference in registers between x86 and x64.	DCWF74 JT0067			
15.3	(U) Describe the ways in which data can move between registers and memory: a. Immediate to register b. Register to register c. Immediate to memory d. Register to memory and vice versa e. Memory to memory	DCWF74 JT0067			
15.4	(U) Describe the purpose of the stack and how data is added/removed from it.	DCWF74 JT0067			
15.5	(U) Describe the following calling conventions: a. cdecl b. stdcall c. fastcall	DCWF74 JT0067			
15.6	(U) Describe purpose and use of foundational RE mechanics: a. Disassembler b. Decompiler c. Debugger	DCWF1062 JT0068 JK0114			
15.7	(U) Describe the difference between static and dynamic analysis.	DCWF1062 JT0068 JK0114			
15.8	(U) Describe how to identify data structures in static analysis.	DCWF1062 JT0068			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
15.9	(U) Demonstrate the ability to reverse engineer a binary and identify key facts about it.	DCWF74 DCWF1062 JT0068 JK0114			
Module 16 - (U) Testing					
16.1	(U) Describe your organization's testing philosophy (e.g. CI/CD, TDD, etc.).	JT0066 JK0110			
16.2	(U) Describe the difference between unit testing and functional/integration testing.	JK0109			
16.3	(U) Describe the difference between modular and monolithic design and its impact on testing.	DCWFN/A			
16.4	(U) Demonstrate the ability to implement a unit test using Python.	DCWFN/A			
16.5	(U) Describe the concept of Continuous Integration (CI) and how it relates to testing and overall product quality.	DCWF756 DCWF1020 A JK0109			
Module 17 - (U) Operations Security (OPSEC)					
17.1	(U) Describe your organization's OPSEC policy.	DCWF1056			
17.2	(U) Describe how to safely use a search engine while maintaining OPSEC.	DCWF1056			
17.3	(U) Demonstrate how to use a search engine without leaking intelligence information.	JS0129			
Module 18 - (U) Embedded Systems					
18.1	(U) Describe how an embedded system differs from a desktop computer and server.	JK0238			
18.2	(U) Describe how the limitations of an embedded system impact development.	JK0238			
18.3	(U) Demonstrate knowledge of alternative C libraries to glibc.	JK0238			
18.4	(U) Demonstrate the ability to cross-compile an application for multiple architectures.	DCWFN/A			
Module 19 - (U) Cyber Capability					

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
19.1	(U) Develop a capability: <ol style="list-style-type: none"> Provide user interface (GUI or CLI) Apply software engineering best practices (CI/CD, virtualization, containerization) Provide documentation for capability 	DCWF23 DCWF1140 A DCWF1071 A DCWF1020 A DCWF973 A DCWF905 DCWF3022 DCWF168 DCWF119 DCWF185 A DCWF3140 DCWF102 JT0059 JT0060 JT0062 JT0074 JT0224 JT0225 JT0081 JK0111 JK0112 JK0108 JK0109 JK0117 JK0116 JA0235 JA0238 JA0239 JA0232 JA0233 JA0234			

Task #	Basic Cyberspace Capability Developer (CCD)	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
End of Basic Cyberspace Capability Developer (CCD) JQS					

(U) Tab 5. (As Required) Service Specific/Operational Training Requirements (SCC, JFHQ-C, JFHQ-DoDIN, CNMF)

(U) The following tasks are additional requirements levied by a component, task force, or team.

Tab 5.1. Service-specific operational training requirements and recommendations

Task #	Service-specific operational training requirements and recommendations	Service/HQ ID#	Trainee Initials	Trainer Initials	Date Qualified
<i>This module covers any Service-specific requirements and recommendations not covered in the preceding modules</i>					
Module 1 -					
1.1	(U) List all unique Knowledge requirements for this work role here; one item per line. Training Resources & Technical References: <ul style="list-style-type: none"> (Enter course title, reading material, etc. here for satisfying JQS item or 'TBD'). List references here for satisfying JQS items 				
End of Service-specific operational training requirements and recommendations JQS					

(U) Tab 6. (Mandatory) Record of assessments, qualifications, Trainer/Mentor/Qualifier/Evaluator memo, formal training completed and recommended training completed

(U) Tab 7. (As Required) Training activity, changes, lapses in training, progress, comments

This page will contain elements of Controlled Unclassified Information (CUI) when completed.

(U) Tab 8. (Mandatory) Signature Page

Trainee

Name: _____

Organization: _____

Signature: _____

Trainer

Name: _____

Organization: _____

Signature: _____

Team Leader/Mentor

Name: _____

Organization: _____

Signature: _____

Approval Date: _____

Qualifier

Name: _____

Organization: _____

Signature: _____

Approval Date: _____