DEPARTMENT OF DEFENSE
UNITED STATES CYBER COMMAND
9800 SAVAGE ROAD, SUITE 6171
FORT GEORGE G. MEADE, MARYLAND 20755-6000

Reply to:
USCYBERCOM J72

April 25, 2023

MEMORANDUM FOR RECORD

Subject: (U) Letter of promulgation for the United States Cyber Command (USCYBERCOM) Basic Capability Developer Job Qualification Record (JQR) v. 1.0

References: (a) CMF Training Pipeline v4.2
(b) Joint Cyberspace Training & Certification Standards v4.0, dated 18 Sep 2020
(c) Cyber Standards Analysis Team (CSAT), 19-21 April 2022

1. (U) Purpose. This memorandum provides guidance and procedures for implementing the joint cyber training and certification standards for the USCYBERCOM Basic Capability Developer JQR.

2. (U) Superseded/Cancellation. This memorandum cancels all previously approved JQRs for the USCYBERCOM Capability Developer at the Basic proficiency level and establishes this as the approved JQR.

3. (U) Applicability. This JQR applies to all personnel operating under CDRUSCYBERCOM authority.

4. (U) Summary of Changes. This JQR has been revised to incorporate all elements of references (a) - (c).

5. (U) Point of Contact. CMF Branch at USCC_J721@cybercom.ic.gov.

6. (U) Effective Date. This memorandum effective the date signed.

David Diana Rose drdavid
Digitally signed by David Diana Rose drdavid
Date: 2023.04.28 14:56:30 -04'00'

DIANA R. DAVID
Chief, Training Division

This content is classified at the **UNCLASSIFIED** level and may contain elements of Controlled Unclassified Information (CUI) or information at a lower classification level than the overall 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 two pages are UNCLASSIFIED** when separated from the rest of this document.This document contains portion markings classified up to the Controlled Unclassified Information (CUI) level.

Controlled by: USCYBERCOM
Controlled By: J72
CUI Categories: INTEL
Limited Dissem. Control: FEDCON, REL TO USA, FVEY
POC: USCYBERCOM/J72, USCC_J721@cybercom.ic.gov

(U) Release History

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

(U) Summary of Changes

VERSION	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 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 unit commanders.

Basic Cyber Mission Force Team JQR

Cyberspace Capability Developer

(U) Introduction to the JQR: Basic 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:

1. **Team Leads** use this JQR to identify the required tasks and KSAs required of a team member performing this work role, and
2. **Training Developers**, who will use the tasks and KSAs to inform the design and development of instruction.

(U) Contents

This JQR is divided into 8 Sections:

- (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 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 Member[1] must have completed all required Training and Certification requirements listed in this JQR. 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) A Cyberspace Capability Developer (CCD) is an innovative, agile, highly skilled practitioner that conducts vulnerability analysis, exploitation research and development, software development, software and user documentation, and implementation of software and hardware capabilities that operates in and through cyberspace. CCDs and their capabilities are foundational elements of cyberspace operations and serve as a force multiplier for the Cyber Mission Force and conventional maneuver forces.

(U) Mission

(U) As part of the Cyber Mission Forces (CMF), the Basic 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

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
Senior	Full Performance
Master	Advanced/Expert

(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 #	USYCBERCOM #	Tasks	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

DCWF #	USYCBERCOM #	Tasks	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
	JT0079	(U) Locate and utilize technical specifications and industry standards (e.g. Internet	Basic

DCWF #	USYCBERCOM #	Tasks	Basic Proficiency Level
		Engineering Task Force (IETF), IEEE, IEC, and International Standards Organization (ISO)).	
	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 #	USYCBERCOM #	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
74		(U) Knowledge of low-level computer languages (e.g., assembly languages).	B
95A		(U) Knowledge of penetration testing principles, tools, and techniques.	A

DCWF #	USYCBERCOM #	KSA	Basic Proficiency Level
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 & 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
3441		(U) Knowledge of physical and logical network infrastructure, to include hubs, switches, routers, firewalls, etc.	B

DCWF #	USYCBERCOM #	KSA	Basic Proficiency Level
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
	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

DCWF #	USYCBERCOM #	KSA	Basic Proficiency Level
	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

DCWF #	USYCBERCOM #	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 Training Requirements

(U) USCYBERCOM does not have a validated training solution for Basic Cyberspace Capability Developer. Service personnel should utilize Service-level training in the interim until a USCYBERCOM-validated training solution is operationalized and added to the CMF Training Pipeline.

(U) JQR Curator-Recommended Training Courses:

(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) SUBSECTION A. 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), 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 JQS lines are mapped to the Basic Cyberspace Capability Developer Tasks and KSAs created in a Cyber Standards Analysis Team (CSAT) 19-21 April 2022 to be codified in the Joint Cyber Training and Certification Standards (JCT&CS). 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.

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
1.0	**(U) Cyberspace Operations Fundamentals** **Training Resources & Technical References:** JP 3-12				
1.1	(U) Describe the mission of the following Cyber Mission Force (CMF) Elements: □ 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: □ Combat Mission Team (CMT) □ National Mission Team (NMT) □ Combat Support Team (CST) □ National Support Team (NST) □ Cyber Protection Team (CPT)	DCWF3146 DCWF3622			
1.3	(U) Read and understand the following cyberspace operations doctrine: □ JP 3-12 (Cyberspace Operations)	DCWF1036 DCWF1151 DCWF3146 DCWF3622 JK0108			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	□ USCYBERCOM Operational Guidance 3-2 (DCO) □ USCYBERCOM Cyber Warfare Publication (CWP 3-33.4)				
1.4	(U) Describe the mission and responsibilities of key organizations: □ U.S. Cyber Command (USCC) □ Director Operation Test and Evaluation (DOTE) □ Federally Funded Research and Development Centers (FFRDC) □ Central Intelligence Agency (CIA) □ Federal Bureau of Investigation (FBI) □ Department of Homeland Security (DHS) □ Defense Threat Reduction Agency (DTRA) □ Defense Digital Service (DDS) □ National Security Agency (NSA) □ National Air and Space Intelligence Center (NASIC)	DCWF320A 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			
2.0	**(U) DCO Fundamentals** **Training Resources & Technical References:** • 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:	DCWF3146			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	□ National CPT □ DoDIN CPT □ Combatant Command CPT □ Service CPT				
2.4	(U) Describe the following DCO work roles: □ Cyber Operations Planner □ Network Analyst □ Host Analyst □ Analytic Support Officer □ Data Engineer □ Network Technician □ 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			
3.0	**(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: □ Remote Operator (ION, RO) □ Mission Commander (MC) □ Exploitation Analyst (EA) □ Digital Network Exploitation Analyst (DNEA) and Target Digital Network Analyst (TDNA) □ Operational Target Development Analyst (OTDA), Targeteer, and Fire Support Planner □ Cyber Operations Planner □ Team Lead and Deputy Team Lead □ Language Analyst (LA) □ Target Analyst Reporter (TAR)	DCWF3146			
4.0	**(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	DCWF40 DCWF414 DCWF515A DCWF3146			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	requirements, and interacting with tool champions, product owners, or requirement-owning stakeholders.	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:** • USCCI 3801-19	DCWF40 DCWF63 DCWF515A JT0076 JK0107 JK0108			
4.3	(U) Describe an example of software development/documentation best practices.	DCWF56 DCWF116 DCWF119 DCWF118 DCWF515A DCWF543 DCWF756 DCWF764 DCWF785 DCWF905 DCWF1151 JT0066 JT0070 JT0075 JT0082 JK0111 JK0118			
5.0	**(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.	DCWF973A JT0070 JT0079 JK0111 JA0233 JA0236			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
5.2	(U) With references, resources, and a provided unit-level coding standard or style guide, identify coding standard violations in C source code.	DCWF973A JT0070 JT0079 JK0111 JA0232 JA0236			
6.0	**(U) C Programming** **Training Resources & Technical References:** • The C Programming Language (Kernighan, Ritchie) • Unix man pages • Primer Plus (Prata)				
6.1	(U) Describe the purpose and use of C programming fundamentals: □ 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: □ 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	DCWF3140			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	related to the implementation of a stack and heap in a C program				
6.3	(U) Demonstrate the proper declaration, understanding, and use of C data types and underlying structures: □ char □ short □ int □ long □ long long □ float □ double □ long double	DCWF3140 JA0232			
6.4	(U) Demonstrate proper declaration, understanding, and use of fixed-width C data types defined in stdint.h: □ int8_t □ uint8_t □ int16_t □ uint16_t □ int32_t □ uint32_t □ int64_t □ uint64_t	DCWF3140 JA0232			
6.5	(U) Demonstrate the ability to create and implement a function that uses different arrays: □ An array □ 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): □ Addition □ Subtraction □ Multiplication □ Division □ Modulus (%) □ Pre-Increment (++i) □ Post-Increment (i++) □ Pre-Decrement (--i) □ Post-Decrement (i--)	DCWF3140 JA0232			

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

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
6.11	(U) Demonstrate skill in using pointers: □ 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: □ 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. □ 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 □ Show or describe the relationship (if any) between each state in the program □ Account for any possible error conditions that may occur during state transitions	JA0234			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
6.15	(U) Describe terms associated with compiling, linking, debugging, and executables: □ Portable Executable (PE) □ Executable and Linkable Format (ELF) □ Difference between PE and ELF □ Difference between a library (shared object / DLL) and a regular executable program □ Calling convention/Application Binary Interface (ABI)	DCWF168 DCWF3140 JA0232			
6.16	(U) Demonstrate skill in compiling, linking, and debugging: □ Execute a program in a debugger to perform general debugging actions □ Create a program using the compilation and linking process □ Compile position-independent code using a cross-compiler	DCWF116 DCWF168 DCWF630 DCWF709A 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: □ and (&) □ or (|) □ xor (^) □ bitwise complement (~) □ shift left (<<) □ shift right (>>) □ Add, removing, and testing for single-bit flags □ Extracting arbitrary bytes from multi-byte data types	DCWF3140 JA0232			
6.19	(U) Demonstrate skill in using the C preprocessor.	DCWF3140 JA0232			
6.20	(U) Demonstrate skill in accessing environment variables.	DCWF3140 JA0232			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
6.21	(U) Demonstrate skill in controlling memory: □ 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	DCWF973A DCWF3140 JA0232			
7.0	**(U) Python Programming** **Training Resources & Technical References:** Python 3 online documentation				
7.1	(U) Describe purpose and use of foundational Python mechanics: □ The return statement □ Data types □ A function □ Parameters □ Scope □ Return values (return type and reference) □ Import files □ Dictionaries □ List □ Tuple □ Singleton □ The term mutable □ The term immutable	DCWF3140 JA0233 JA0236			
7.2	(U) Demonstrate the proper declaration and use of Python data types and object-oriented constructs: □ Integer (int) □ Float (float) □ String (str) □ List (list)	DCWF3140 JA0233			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	□ Multi-dimensional list □ Dictionary (dict) □ Tuple (tuple) □ Singleton				
7.3	(U) Demonstrate the ability to perform basic arithmetic operations using Python operators while ensuring proper order of operations (PEMDAS): □ Addition □ Subtraction □ Multiplication □ Division □ Modulus	DCWF3140 JA0233			
7.4	(U) Demonstrate the ability to perform file management operations in Python: □ Open an existing file □ Read data from a file □ Parse data from a file □ Write data to a file □ Modify data in a file □ Close an open file □ Print file information to the console □ Create a new file □ Append data to an existing file □ Delete a file □ Determine the size of a file □ Determine location within a file □ Insert data into an existing file	DCWF3140 DCWF6780 JA0233			
7.5	(U) Demonstrate the ability to create and implement functions to meet a requirement: □ A function that returns multiple values □ A function that receives input from a user □ A recursive function	DCWF3140 JA0233			
7.6	(U) Demonstrate the ability to perform data validation: □ Validating received input matches expected input □ Designing and implementing a scheme for exception handling	DCWF185A DCWF3140 JA0233			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
7.7	(U) Demonstrate skill in creating and implementing conditional statements, expressions, and constructs: □ for loop □ while loop □ with statement □ if statement □ if/else statement □ if/elif/else statement □ try/except/finally	DCWF3140 JA0233			
7.8	(U) Describe the terms and fundamentals associated with object oriented programming using Python: **Training Resources & Technical References:** *Design Patterns: Elements of Reusable Object-Oriented Software (Gamma, Helm, Johnson, Vlissides)* □ Class □ Object □ Difference between an object when discussing a class □ Advantages to object-oriented programming □ Inheritance □ The keyword ”super” □ Initialization function of a constructor □ The keyword ”self” □ The getter and setter functions □ Attributes of a class □ Factory design pattern □ Singleton design pattern □ Adapter design pattern □ Bridge design pattern	DCWF23 DCWF119 DCWF3140 DCWF6780 JA0233			
7.9	(U) Demonstrate the ability to parse command line arguments using built-in functionality.	DCWF3140 JA0233			
8.0	**(U) Data Structures** **Training Resources & Technical References:** *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:	DCWF102 JT0073 JK0117			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	□ 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)	JA0239			
8.2	(U) Demonstrate skill in creating and using a circularly linked list that accepts any data type: □ Creating a circularly linked list with n number of items □ Navigating through a circularly linked list □ Finding the first occurrence of an item in a circularly linked list □ Sorting the circularly linked list alphanumerically using a function pointer □ Removing selected items from the circularly linked list □ Inserting an item into a specific location in a circularly linked list □ Removing all items from the circularly linked list □ Destroying a circularly linked list	DCWF102 JT0073 JK0117 JA0239			
8.3	(U) Demonstrate skill in creating and using a binary search tree that accepts any data type: □ Creating a binary search tree with n number of items □ Navigating through a binary search tree □ Locating an item in a binary search tree □ Removing selected items from the binary search tree □ Removing all items from the binary search tree □ Describe implementation strategies for a balanced binary search tree □ Destroying a binary search tree	DCWF102 JT0073 JK0117 JA0239			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
8.4	(U) Demonstrate skill in creating and using a hash table that accepts any data type: □ Creating a hash table with n number of items □ Navigating through a hash table to find the nth item □ Finding an item in a hash table □ Removing selected items from a hash table □ Inserting an item into a hash table □ Implement functionality to mitigate hash collisions within the hash table □ Removing all items from the hash table	DCWF102 JT0073 JK0117 JA0239			
8.5	(U) Demonstrate skill in creating and using a stack that accepts any data type: □ Create a stack (cannot be fixed sized) □ Adding an item in a stack (enforce FILO) □ Removing n items from a stack □ Removing all items from the stack □ Destroying a stack □ Preventing a stack overrun	DCWF102 JT0073 JK0117 JA0239			
8.6	(U) Demonstrate skill in implementing a priority queue that accepts any data type: □ Defining the underlying structures required for priority queues (cannot be fixed sized) □ Assigning a priority to each element □ Inserting an element into the priority queue □ Removing the element with the highest priority from the priority queue □ Destroying a priority queue □ Define possible applications of a priority queue	DCWF102 JT0073 JK0117 JA0239			
9.0	**(U) Algorithms**				
9.1	(U) Describe concepts associated with traversal techniques: □ Depth first traversal □ Breadth first traversal □ The technique of determining the weight of a given path when traversing a graph □ How the most efficient path for traversing a graph is determined	DCWF102 DCWF119 JT0072 JK0117			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
9.2	(U) Describe concepts associated with hashing: □ Data distribution as it relates to hashing □ Hash function efficiency □ Hash collisions	DCWF102 DCWF119 JT0072			
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 □ Insertion sort □ Selection sort □ Merge sort □ Heap sort □ Quick sort □ Hashing	DCWF102 DCWF119 JT0072 JT0073 JK0117 JA0239			
10.0					
10.1	(U) Describe terms and concepts associated with Operating System (OS) virtualization: □ Processes □ CPU scheduling □ Paging tables □ Caching □ Kernel and user-mode memory	JK0112			
10.2	(U) Describe the following terms and concepts: □ File systems □ The boot process	JK0112			
10.3	(U) Demonstrate the ability to use the following constructs: □ Interrupts □ Signal handling	JK0112			
10.4	(U) Describe terms and concepts associated with concurrency: □ Threading (thread vs pthread) □ fork □ join □ create □ exit □ detach □ self □ Locking (mutex, semaphore, etc) □ Race conditions □ Deadlocks	JK0112			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
	□ thread safe □ thread id □ conditional variables □ atomics □ 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: □ Threads □ Locks □ Condition variables □ Atomics □ Thread Pool (with graceful shutdown without memory leaks)	JK0112			
11.0					
11.1	(U) Describe terms and concepts associated with secure coding practices: □ Common string-handling functions □ Which functions guarantee null terminated strings □ An off-by-one error □ An integer overflow □ A buffer overflow □ The concept of use-after-free □ Resource acquisition is initialization (RAII) □ The difference between input validation vs. input sanitization □ The meaning of a pure function and if a function has a side-effect □ 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) □ Penetration testing principles, tools, and techniques □ Obfuscation techniques	DCWF27 DCWF56 DCWF95A DCWF905 DCWF1071A JT0076 JT0227 JK0298			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
11.2	(U) Demonstrate skill in using secure coding techniques: □ Formatting string vulnerabilities □ Safe buffer size allocation □ Input sanitization □ Input validation □ Establish a secure communications channel using an SSL library □ Securely zeroing-out memory (compiler optimizations)	DCWF543 DCWF764 DCWF905 DCWF1071A DCWF1140A DCWF1149A JT0075 JT0076			
12.0					
12.1	(U) Describe the concepts and terms associated with networking fundamentals: □ Transmission Control Protocol (TCP) / User Datagram Protocol (UDP) □ Open Systems Interconnect (OSI) model □ POSIX API/BSD sockets □ Purpose and use of sockets □ Request For Comments (RFCs) □ Purpose of subnetting	DCWF22			
12.2	(U) Describe the concepts and terms associated with common protocols and their associated ports, if applicable: □ Address Resolution Protocol (ARP) □ Hypertext Transfer Protocol/Secure (HTTP/HTTPS) □ Domain Name System (DNS) □ Simple Mail Transfer Protocol (SMTP) □ Internet Control Message Protocol (ICMP) □ Dynamic Host Configuration Protocol (DHCP) □ Internet Protocol version 4 (IPv4) □ Internet Protocol version 6 (IPv6)	DCWF22			
12.3	(U) Describe the addressing associated with key networking protocols: □ IPv4 □ IPv6 □ Ethernet	DCWF22			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
12.4	(U) Describe the concepts and terms associated with physical and logical network infrastructure: □ Hubs □ Switches □ Routers □ Firewalls	DCWF22 DCWF3441			
12.5	(U) Describe different types of network communications: □ LAN □ WAN □ MAN □ WLAN □ WWAN	DCWF22 DCWF278			
12.6	(U) In Python, demonstrate skill in using networking commands accounting for endianness: □ socket() □ send() □ recv() □ sendto() □ recvfrom() □ bind() □ listen() □ connect() □ accept() □ close() □ gethostname()	DCWF22 JA0238			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
12.7	(U) In C, demonstrate skill in using networking commands accounting for endianness: □ socket() □ send() □ recv() □ sendto() □ recvfrom() □ bind() □ connect() □ accept() □ getsockopt() □ setsockopt() □ getaddrinfo() □ gethostname() □ struct sockaddr □ struct sockaddr_in □ struct sockaddr_un □ 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 □ select □ poll	DCWF22			
13.0					
13.1	(U) Describe data serialization and de-serialization.	JT0072 JK0116			
13.2	(U) Demonstrate the ability to handle partial reads and writes during serialization and de-serialization.	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			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
13.4	(U) Demonstrate the ability to serialize and de-serialize 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			
14.0					
14.1	(U) Describe the following concepts related to regular expressions: □ Basic Regular Expressions (BRE) □ Extended Regular Expressions (ERE) □ Perl Compatible Regular Expressions	DCWF3140			
14.2	(U) In C, demonstrate the ability to incorporate regular expression processing into a program □ Matching □ Use of capture groups	JA0232			
14.3	(U) In Python, demonstrate the ability to incorporate regular expression processing into a program □ Matching □ Use of capture groups	JA0233			
15.0					
15.1	(U) Describe the purpose of the following registers: □ General Purpose Registers □ EIP □ 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: □ Immediate to register □ Register to register □ Immediate to memory □ Register to memory and vice versa □ Memory to memory	DCWF74 JT0067			
15.4	(U) Describe the purpose of the stack and how data is added/removed from it.	DCWF74 JT0067			

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
15.5	(U) Describe the following calling conventions: □ cdecl □ stdcall □ fastcall	DCWF74 JT0067			
15.6	(U) Describe purpose and use of foundational RE mechanics: □ Disassembler □ Decompiler □ 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			
15.9	(U) Demonstrate the ability to reverse engineer a binary and identify key facts about it.	DCWF74 DCWF1062 JT0068 JK0114			
16.0					
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.	N/A			
16.4	(U) Demonstrate the ability to implement a unit test using Python.	N/A			
16.5	(U) Describe the concept of Continuous Integration (CI) and how it relates to testing and overall product quality.	JK0109 DCWF756 DCWF1020A			
17.0					
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			
18.0					

Task #	Basic Cyberspace Capability Developer Knowledge	JCT&CS ID#	Trainee Initials	Trainer Initials	Date Qualified
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.	N/A			
19.0	**(U) Cyber Capability**				
19.1	(U) Develop a capability: □ Provide user interface (GUI or CLI) □ Apply software engineering best practices (CI/CD, virtualization, containerization) □ Provide documentation for capability	DCWF23 DCWF102 DCWF119 DCWF168 DCWF185A DCWF905 DCWF973A DCWF1020A DCWF1071A DCWF1140A DCWF3022 DCWF3140 JT0059 JT0060 JT0062 JT0074 JT0081 JT0224 JT0225 JK0108 JK0109 JK0111 JK0112 JK0116 JK0117 JA0232 JA0233 JA0234 JA0235 JA0238 JA0239			

(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.

Task #	Basic Cyberspace Capability Developer Planner Knowledge	Service/HQ ID#	Trainee Initials	Trainer Initials	Date Qualified
1.0					
<Enter item number here (i.e. 1.1)>	(U) *List all unique Knowledge requirements for this work role here; one item per line.* Training Resources & Technical References: (*(Enter course title, reading material, etc. here for satisfying JQS item or 'TBD'). List references here for satisfying JQS items*				

(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: _____