CYBERSECURITY PEDAGOGY & PRACTICE JOURNAL

Volume 5, No. 1
April 2026
ISSN: 2832-1006

In this issue:

iscap
Information Systems & Computing Academic Professionals

The **Cybersecurity Pedagogy and Practice Journal** (**CPPJ**) is a double-blind peer-reviewed academic journal published by **ISCAP** (Information Systems and Computing Academic Professionals). Publishing frequency is two times per year. The first year of publication was 2022.

CPPJ is published online (https://cppj.info). Our sister publication, the proceedings of the ISCAP Conference (https://proc.iscap.info) features all papers, panels, workshops, and presentations from the conference.

The journal acceptance review process involves a minimum of three double-blind peer reviews, where both the reviewer is not aware of the identities of the authors and the authors are not aware of the identities of the reviewers. The initial reviews happen before the ISCAP conference. At that point, papers are divided into award papers (top 15%), and other accepted proceedings papers. The other accepted proceedings papers are subjected to a second round of blind peer review to establish whether they will be accepted to the journal or not. Those papers that are deemed of sufficient quality are accepted for publication in the CPPJ journal.

While the primary path to journal publication is through the ISCAP conference, CPPJ does accept direct submissions at https://iscap.us/papers. Direct submissions are subjected to a double-blind peer review process, where reviewers do not know the names and affiliations of paper authors, and paper authors do not know the names and affiliations of reviewers. All submissions (articles, teaching tips, and teaching cases & notes) to the journal will be refereed by a rigorous evaluation process involving at least three blind reviews by qualified academic, industrial, or governmental computing professionals. Submissions will be judged not only on the suitability of the content but also on the readability and clarity of the prose.

Currently, the acceptance rate for the journal is under 35%.

Questions should be addressed to the editor at editorcppj@iscap.us or the publisher at publisher@iscap.us. Special thanks to members of ISCAP who perform the editorial and review processes for CPPJ.

CYBERSECURITY PEDAGOGY AND PRACTICE JOURNAL

Editors

2026 Review Board

Excessive Equating: An Exploration of Knowledge Unit (KU) Curricular Load for CAE-CD Program Design and Evaluation

Kasey Miller
millerkc@uncw.edu

Kevin Matthews
matthewskd@uncw.edu

Ulku Clark
clarku@uncw.edu

Geoff Stoker
stokerg@uncw.edu

Congdon School
University of North Carolina Wilmington
Wilmington, NC 28403 USA

Abstract

The growing demand for rigorous, standardized cybersecurity education has made the NSA's National Centers of Academic Excellence in Cybersecurity (NCAE-C) program a cornerstone in ensuring quality and consistency across institutions. The NCAE-C program for Cyber Defense utilizes the fundamental element Knowledge Unit (KU) to bundle learning outcomes and topics. Institutions designated a Center of Academic Excellence (CAE) under the NCAE-C program must validate at least one program of study (PoS) by mapping PoS courses to a specified number and set of KUs. This ensures that the CAE's PoS includes foundational cybersecurity content and provides sufficient breadth and depth. A simplifying NCAE-C program guideline treats all KUs as equivalent for mapping and validation purposes, regardless of the number or difficulty of learning outcomes and topics. In this paper, we suggest that a more granular approach may be appropriate when comparing KUs. Using systematic counts of learning outcomes and topics, combined with Bloom's Taxonomy weighting of cognitive verbs, we calculate curricular load scores for all 73 KUs in the CAE-CD program. These findings suggest that uniform treatment of KUs may unintentionally introduce inequities into CAE-CD program design, review, and evaluation. Recommendations include standardizing verb usage across KU documents and considering KU complexity when developing or revising program criteria. A more granular understanding of KU demands can enhance curriculum planning and strengthen both academic rigor and alignment with evolving cybersecurity needs.

Keywords: Knowledge Unit (KU), CAE-CD, Bloom, Cybersecurity Pedagogy

Recommended Citation: Miller, K., Matthews, K., Clark, U.Y., Stoker, G., (2025). Excessive Equating: An Exploration of Knowledge Unit (KU) Curricular Load for CAE-CD Program Design and Evaluation. *Cybersecurity Pedagogy and Practice Journal;* v5(1) pp 17-40. DOI# https://doi.org/10.62273/SYPT8785

Excessive Equating: An Exploration of Knowledge Unit (KU) Curricular Load for CAE-CD Program Design and Evaluation

Kasey Miller, Kevin Matthews, Ulku Clark, Geoff Stoker

1. INTRODUCTION

One of the two main requirements for designation in the National Centers of Academic Excellence in Cybersecurity (NCAE-C) program for Cyber Defense (CD) is a validated Program of Study (PoS) (Application Process and Adjudication Rubric Cyber Defense Working Group, 2024). A major part of validating a PoS for a bachelor's program involves aligning 22 knowledge units (KU) with relevant courses within the PoS [NOTE: KU alignment details differ across associate, master's, and doctoral programs]. "A Knowledge Unit (KU) is a thematic grouping that encompass [*sic*] multiple, related KU outcomes and learning topics." (Application Process and Adjudication Rubric Cyber Defense Working Group, 2024, p. 3). In this paper, the term "curricular load" refers to an abstract measure of the academic burden associated with a set of learning outcomes and topics. Although this concept is explained more fully later, for now, think of curricular load as the idea that covering one learning outcome is less demanding than covering two, and addressing one topic is less burdensome than addressing two.

Currently, there are 73 KUs grouped as follows:

- 3 Foundational KUs
- 5 Technical Core KUs
- 5 Non-technical Core KUs
- 60 Optional KUs.

Each validated PoS in a bachelor's degree program must align with 1) the 3 Foundational KUs, 2) *either* all 5 Technical Core KUs *or* all 5 Non-technical Core KUs, and 3) 14 of the Optional KUs (*NOTE: opposing core KUs may also be used as optional KUs – i.e., if the Technical Core is chosen, then Non-technical Core KUs may be used as optional KUs, and vice versa*). Each KU contains a list of learning outcomes and a list of topics. "While it is not required that every learning outcome be explicitly assessed as written, applicant schools should be able to defend their coverage of the learning outcomes" (Becker et al, 2024, p. 3). For KU topics coverage, a simple majority must be addressed.

The NCAE-C Program for CD instruction document also specifies that "a KU may be covered by one or more courses; however, a course should not be aligned to an excessive number of KUs, given the challenge of so many KU Outcomes coverage with a single course" (Becker et al, 2024, p. 3). The meaning of *excessive* is not clarified in this document, but in recent guidance from the NCAE-C program office, the number five has been suggested as the threshold above which mapped KUs to a single course would be scrutinized (S. Steiner, personal communication, May 22, 2025).

While this guidance begins to clarify what excessive could mean and is administratively useful, it is a bit coarse-grained and seems to imply, likely unintentionally, that the curricular load of all KUs is equivalent, so 1 KU ≡ 1 KU, despite the variation in the number of learning outcomes and topics for each KU. Among the 73 KUs, the number of learning outcomes per KU ranges from 1 to 10, and the number of topics ranges from 5 to 41. At the extremes, the KU Software Security Analysis (SSA) has 2 Learning Outcomes and 5 Topics, whereas the KU Hardware/Firmware Security (HFS) has 5 Learning Outcomes and enumerates 41 Topics.

Although CAE CD policy treats all KUs as equivalent, the number of learning outcomes, topics, and Bloom-level verbs varies dramatically. Without accounting for this variation, KU mapping may unintentionally penalize programs whose selected KUs carry disproportionately high cognitive or topical load.

This observation raises some questions, the exploration of which seems likely to be beneficial to the CAE-CD community. Specifically, what is a good way to assess the curricular load of a particular KU? Would having a curricular load score for each KU be helpful when evaluating a school's PoS? Would a curricular load score help schools interested in applying to the NCAE-C program better align KUs to their curriculum?

In this paper, two ideas for generating a KU curricular load score using the number of KU Learning Outcomes, the number of KU Topics, and the revised Bloom's Taxonomy level associated with the measurable verbs in the KU Learning Outcomes are explored. Section 2 reviews Bloom's Taxonomy very lists and prior KU

analysis to motivate why verb choice matters for curricular burden. Section 3 details two scoring techniques, unweighted curricular load score (UCLS) and weighted curricular load score (WCLS), and the coding protocol used. Section 4 reports the results across KUs and illustrates the differences between UCLS and WCLS. Section 5 interprets the results for academic units from course design and program evaluation perspectives. Section 6 concludes with implications for standards-aligned curricula beyond CAE-CD.

2. LITERATURE REVIEW

In 1948, an informal meeting of college examiners sparked interest in creating a theoretical framework to facilitate communication and the exchange of assessment items across educational institutions that measure common educational objectives (Bloom et al., 1956; Krathwohl, 2002). The original idea included plans for a taxonomy of three domains: *cognitive*, *affective*, and *psychomotor*. After years of work, a handbook was published on the cognitive domain, focusing on "the recall or recognition of knowledge and the development of intellectual abilities and skills" (Bloom et al., 1956, p. 7). The six major classes identified were: *knowledge*, *comprehension*, *application*, *analysis*, *synthesis*, and *evaluation*.

About half a century later, the framework was revised by a group that included David R. Karthwohl, a key contributor and author of the original framework, and resulted in the renaming of three classes, the reordering of two, and the recasting of all to verb form: *remember*, *understand*, *apply*, *analyze*, *evaluate*, and *create* (Krathwohl, 2002).

Verbs

Using the presence of specific verbs in learning objectives to help identify and map objectives to Bloom levels has been done since the 1956 publication of the original taxonomy; however, an authoritative, non-level-overlapping list of verbs does not currently exist. Several efforts have been made to curate such a list, and we explore here the five that we consulted.

Thirty unique verb lists were gathered by Stanny (2016) from the top 30 results of a Google search for "action words for Bloom's taxonomy" (Stanny, 2016, p. 3). From this collection of 788 verbs, she found 433 unique verbs and 355 duplicates, both within and across the six Bloom categories. Using frequency of appearance across the 30 lists, Stanny created a list of 104 unique verbs that each appeared on 10 or more lists. These 104 verbs resulted in a 128-verb chart with 18 words duplicated across Bloom categories and the triplication of three (Figure A-1).

Newton et al. (2020) gathered 47 publicly available lists from 35 universities and textbooks, noting that there was "very little agreement between these lists, most of which were not supported by evidence explaining where the verbs came from" (Newton et al., 2020, p. 1). Across the lists, they found 401 unique verbs. They created a 51-verb list with no duplicates using the original Bloom categories. It included only verbs that appeared on more than half of the lists, occurring 50% of the time in one category (Figure A-2).

In 2022, Das et al. built upon Stanny's work and created a four-level classification system: Level 1, unambiguous; Level 2, unambiguous with a lower threshold; Level 3, transitional verbs; and Level 4, ambiguous. Level 1 results in 83 verbs, which is Stanny's 128-verb chart minus the 21 verbs that repeat (Figure A-3).

In January 2023, the Association for Computing Machinery (ACM) Committee for Computing Education in Community Colleges (CCECC) published a report that included a chart with 142 unique verbs (Bamkole et al, 2023). While many of the verbs are common to lists found on the internet, the main purpose of the report was to curate verbs useful to the computing community and for "technical tasks for which a technical verb would be appropriate but is not available" (Bamkole, 2023, p. 5). For example, they took the verbs *code*, *script*, and *program*, which indicate similar concepts, and assigned *code* to the Apply level and *script* and *program* to the Create level. The published list includes 56 of these compute-related verbs (Figure A-4).

For their 2024 article, ElJishi et al. obtained lists of action verbs aligned to the revised Bloom's Taxonomy from Stanford, Harvard, and an open textbook by Zhou & Brown (2015). They used consensus to avoid duplicating verbs across Bloom categories and created a 140-verb list, albeit with 4 duplicates (Figure A-5).

KU Analysis

Previous analysis evaluated the 2018-2019 changes to KU mapping and the reorganization of KU groups from two-year core, four-year core, and optional to the current groups of foundational, technical core, non-technical core, and optional (Clark et al., 2020). This paper considers the KUs with changes published in late

2024 and focuses on their learning objectives and topics.

3. METHODOLOGY

Curricular load scores for each of the 73 KUs were generated in two ways: an unweighted method and a weighted method incorporating the revised Bloom's taxonomy levels. A complicating factor for both methods was how to count topics in the 36 KUs with enumerated subtopics. In these cases, only subtopics were counted, and the topic was treated as a heading. In Figure 1, for example, the Technical Core KU Basic Scripting and Programming (BSP) has eight numbered topics, one of which (number 8) includes 10 subtopics enumerated by lowercase letters a. through j. In this case, we count 17 topics for the BSP KU (7 topics + 10 subtopics).

Topics

1. Basic security concepts
2. Permissions (e.g., Linux, Windows, MacOS), bounds checking, input validation, type checking and parameter validation
3. Fundamental concepts and basic implementation of regular expressions
4. Fundamental data structures and algorithms
5. Boolean logic/operations (e.g., AND / OR / XOR / NOT)
6. Scripting language on both Windows and Linux (e.g. PERL, Python, BASH, JAVA, VB Scripting, Powershell)
7. Integrated Development Environment (IDE), Compilers/Interpreters
8. Properly apply basic programming constructs and concepts including:
    a. Variables and types (e.g., int, float, char, etc.)
    b. Strings, arrays, structures
    c. Sequential and parallel execution
    d. Assignments (e.g., :=, =, ++, --, etc.)
    e. Decisions and branching (e.g., if, if ... else, elseif, switch, case, etc.)
    f. Loops (e.g., for, while, repeat, etc.)
    g. Functions, procedures, and calls
    h. Debugging techniques
    i. Console and file I/O
    j. Libraries

Figure 1: BSP KU topics count = 17 – topics 1-7 plus subtopics 8.a.-8.j. (Becker et al., 2024, p. 20).

For Information Assurance Compliance (IAC), this one KU with sub-subtopics (Becker et al., 2024, p. 73) followed the same guideline. In this case, only the sub-subtopics were counted; the topic and subtopic were treated as headings.

Unweighted Curricular Load Scores

The unweighted curricular load score (UCLS) is simply a count of the enumerated learning outcomes and listed topics for each KU. For example, the KU Systems Certification and Accreditation (SCA) has 2 numbered learning outcomes and 5 numbered topics (Figure 2), so the UCLS for SCA is 7 (i.e., 2 + 5).

Weighted Curricular Load Scores

The weighted curricular load score (WCLS) calculation involves an additional step: weighting each learning outcome. Instead of a value of 1, as with UCLS, each learning outcome is given a value (weight) from 1 to 6 based on the Bloom's Taxonomy category into which the verb maps. The SCA KU (Figure 2) learning outcome #1 verb, *describe*, maps to Bloom's *understand* tier (level 2), and the learning outcome #2 verb, *define*, maps to Bloom's *remember* tier (level 1). The weighted score for the SCA KU topics is 2 + 1 = 3, and the WCLS is 8 (3 weighted topic score + 5 topics).

KU Learning Outcomes

To complete this KU, students will be able to:

1. Describe the DoD system certification and accreditation processes.
2. Define certification and accreditation.

Topics

1. DoD Policies and Directives
2. Roles/Players
3. Components of the C&A Process
4. Certification Boards and Panels
5. NIST Risk Management Framework (SP800-37)

Figure 2: Learning Outcomes and Topics for the SCA KU.

For learning outcomes with a single verb, the mapping is straightforward. For learning outcomes with more than one verb, we map it to the Bloom level of the highest-order verb. For example, learning outcome #2 for the Optional KU Data Administration (DBA) reads: "Define and evaluate data and information quality, accessibility, and utility" (Becker et al, 2024, p. 53). This learning outcome has two action verbs: *define* and *evaluate*. Define maps to Bloom's *remember* tier (level 1) and evaluate maps to the *evaluate* tier (level 5), so this learning outcome would have a weight of five.

Results for all UCLS and WCLS values for each KU are provided in detail in Tables B-2 and B-3 and depicted graphically in Figures B-1 through B-3.

Counting and Coding

To identify potential errors or omissions in reviewing the KUs, three authors independently reviewed the KU document, focusing on each of the 73 KUs' learning outcomes and topics. A spreadsheet that captured three things for each KU was produced by each author:

- count of the learning outcomes
- count of the topics
- verb(s) in each learning outcome

After all 73 KUs were coded by each author, the results were compared. All three coders met to review discrepancies and unanimously agreed on the correct code for each disagreement.

First, the codes for the count of learning outcomes were reviewed, and one KU showed a

disagreement (0.014%). Upon further review, one coder mistakenly swapped the values for the learning outcome and topic counts for this KU.

Next, the codes for topic counts were reviewed, and disagreements were identified in 14 KUs (19.178%). Upon further review, there were two main categories of coding disagreements: formatting issues in the Becker et al. (2024) document and human error during the coding process. Only one disagreement fell outside these categories and could not be explained. Nevertheless, all disagreements were easily resolved with unanimous agreement among all three coders.

The formatting issues in the Becker et al. (2024) document accounted for 6 disagreements and can be broken down into 3 types. Four disagreements occurred because a page break separated the enumerated topics, and one coder missed the orphaned topics on the following page (e.g., p. 10). One disagreement occurred because the KU topics list was missing a line break, and the final topic was included on the same line as the previous subtopic (p. 73). The final disagreement was over the inclusion of three "examples of acceptable operating system specific Topics" for the Host Forensics (HOF) KU (p. 71). These operating system-specific examples were ultimately determined to be extensions of previous topics that were already counted in that KU. While this is not necessarily a "formatting" issue with the document, it was the only KU that had such a supplemental list.

Human errors led to seven coding disagreements and can be broken down into four types. One disagreement, paired with the learning outcome disagreement, where the coder swapped the counts of the learning outcomes and the topics. Three disagreements occurred because one or more coders did not include subtopics in a KU's topic count. Two disagreements occurred due to typos: a coder prepended a 1 to the count (i.e., 19 instead of 9 and 15 instead of 5). And finally, one disagreement occurred because a coder mistakenly coded a KU in the wrong row of the spreadsheet; that is, they coded the previous KU instead of the current KU. These types of errors were located, resolved, and support our decision to have multiple coders.

The final thing to review was the verb(s) in each learning outcome. For all 73 KUs, there were 293 learning objectives. Of these 293 learning objectives, 94 (32.082%) contained more than 1 action verb and required a decision of which verb had the highest Bloom level. After all coding was completed, there were disagreements with 36 of the 293 verbs selected (12.287%).

Disagreement over verbs required a bit more discussion among the coders than did the counts of learning outcomes and topics. Once disagreements were identified, all three coders reviewed discrepancies together, adjusted the verb selection process as needed, and unanimously agreed on the selected verb. The reconciliation process revealed a few trends in the discrepancies. These trends were all rooted in the interpretation of the learning objectives and in the ability to reliably reach the same conclusion for a selected verb. In its simplest form, the disagreement was over the order of the listed verbs. For example, if a learning objective contained two verbs at the same Bloom level, sometimes the coders selected the verb that occurred first in the sentence, while other times they selected the alphabetically ranked verb. For consistency, the verb written first in the learning objective was selected.

Surprisingly, the verb selection process also involved parsing the learning objective to identify verb candidates and ruling out verbs that were merely supplemental to the learning objective's action. For example, learning objective #2 for Basic Networking (BNW) reads, "Apply networking concepts to design a basic network architecture given a specific need and set of hosts/clients" (Becker et al., 2024, p. 18). While all coders identified "apply" as an action verb, they differed in how they treated the word "design." After discussion, it was agreed that "to design" was supplemental to the primary action verb "apply" and that this and any subsequent constructions of "to [verb]" would be treated similarly. The same rule was also applied to a sentence with the construction "to [verb1] and [verb2]", reading "verb2" as having an implied "to" just before it. For example, Media Forensics (MEF), learning objective #2 reads, "Apply forensics techniques to investigate and [to] analyze a particular media in context" (Becker et al., 2024, p. 84).

Another interesting discrepancy arose with learning objective #2 for the Optional KU Network Forensics (NWF), which reads, "Analyze and decipher network traffic, identify anomalous or malicious activity, and provide a summary of the effects on the system" (Becker et al., 2024, p. 86). In this case, one author coded "provide a summary" as "summarize" instead of "provide." Though there was general agreement that "summarize" was probably a better verb for that learning objective, coding was restricted to the

document's original text only.

Mapped Verbs

When determining which verbs map to which Bloom levels, the study relied heavily on previous efforts to curate consensus lists (Bamkole et al., 2023; Das et al., 2022; ElJishi et al., 2024; Newton et al., 2020; Stanny, 2016). Each unique verb from the KUs was placed into a Bloom category by referencing the lists in Figures A-2 through A-5. If the verb was in the same category in all 4 lists, placement was easy. If a verb was missing in one or more lists, and the remaining lists had the verb in the same category, placement was also easy. For conflicting listings, we developed the following rules to help us place verbs:

- If 3 of 4 or 2 of 3 lists agreed, the majority ruled
- If 1-1 or 2-2 tie, default to the ACM list
- If 1-1 tie with no ACM or no list had the verb, the verb was placed using the researchers' judgment

The final verb list and Bloom-level categorizations are shown in Table 1. Of the 70 unique verbs across all 73 KUs' learning outcomes, complete consensus mapping was found for 14 (20%) of the verbs (***bold/italics*** Table 1) and some degree of consensus for an additional 20. A single source was used to map 16 verbs. For 20 verbs, categorization was based on the researchers' judgment because none of the lists contained those verbs, or there was conflicting Bloom-level alignment across two lists that were not the ACM list. The details of the placement results are reflected in Table B-1.

Unweighted Technique

Using the unweighted method, it is found that the Systems Certification and Accreditation (SCA) KU was the most lightweight with a UCLS of 7 (2 learning outcomes + 5 topics), and the Hardware/Firmware Security (HFS) KU was the most heavyweight with a UCLS of 46 (5 learning outcomes + 41 topics). A list of all KUs ordered by UCLS is provided in Table B-2.

Weighted Technique

Using the weighted method, it is found that the Systems Certification and Accreditation (SCA) KU was still the most lightweight with a WCLS of 8 – 2 learning outcomes: (1 * level 1 + 1 * level 2 = 3) + 5 topics – while the Intrusion Detection/Prevention Systems (IDS) KU became the most heavyweight with a WCLS of 55 – 7 learning outcomes: (5 * level 3 + 1 * level 4 + 1 * level 5 = 24) + 31 topics. A list of all KUs ordered by WCLS is provided in Table B-3. A list of all KUs, ordered by verb weight, with the corresponding learning outcome verbs used for the weighting process when calculating the WCLS, is provided in Table B-4. A list of the 94 multiple-verb KU learning outcomes is provided in Table B-5, with the verbs not used in the weighting calculation identified.

Remember
define, identify, ***list***, ***recall***, recognize, select
Understand
annotate, communicate, describe, ***discuss***, ***explain***, explore, review, understand
Apply
apply, assist, compute, conduct, configure, demonstrate, deploy, document, draw, execute, handle, harden, illustrate, implement, install, leverage, map, mitigate, perform, produce, protect, provide, quantify, ***use***, utilize
Analyze
analyze, articulate, categorize, characterize, compare, ***contrast***, decipher, detect, diagram, differentiate, examine, monitor, outline, resolve
Evaluate
assess, determine, ***evaluate***, rate, recommend, set up, suggest, test
Create
create, ***design***, develop, devise, organize, ***plan***, propose, prototype, write

Table 1: 70 unique verbs across the 73 KUs mapped to the Revised Bloom's Taxonomy. Verbs in bold indicate complete consensus for mapping across all source verb lists.

4. RESULTS

Verbs

From the 402 measurable verbs used across the 73 KUs' 293 learning outcomes, 70 unique verbs were found. The verb *describe* was used 91 times (22.6% of the 402). There were 34 verbs used a single time (Table 2), for a total of 8.5% of the 402 verb uses. The six verbs *describe*, *apply*, *explain*, *identify*, *understand*, and *evaluate* account for 50.2% of all verb uses. Verb frequency information is available in Table B-6.

articulate, assist, categorize, characterize, communicate, compute, conduct, decipher, detect, devise, diagram, document, draw, explore, handle, harden, illustrate, map, mitigate, monitor, organize, produce, protect, prototype, quantify, rate, recognize, resolve, review, select, set up, suggest, test, utilize

Table 2: verbs used only a single time

5. ANALYSIS AND DISCUSSION

The main benefit of this analysis is that the range

of academic burden across the 73 KUs becomes evident when viewed through the lens of curricular load scores. This begins to make clear why it might be worthwhile considering an alternative to 1 KU ≡ 1 KU.

The small graph in Figure 3 provides a sense of how the UCLS differs across all KUs, from the most lightweight KU, Systems Certification and Accreditation (SCA), with the fewest learning outcomes and topics and a UCLS of 7, to Hardware/Firmware Security (HFS), the KU with the most learning outcomes and topics and a UCLS of 46. A larger version of this graph is provided in Figure B-1.

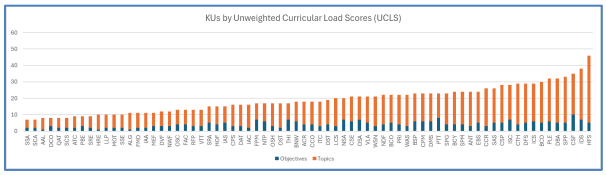

Figure 3: KUs by UCLS

Further comparisons of the KUs using WCLS, as in Figure 4, suggest that the academic burden difference among KUs is likely even greater. The weighting of learning outcomes reveals subtle differences among the KUs and shifts the ordering. While SCA remains the least complex KU with a WCLS of 8, Intrusion Detection/Prevention Systems (IDS) emerges as the most complex with a WCLS of 55. A larger version of this graph is provided in Figure B-2.

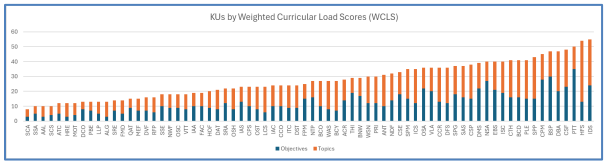

Figure 4: KUs by WCLS

Since topics are treated the same for UCLS and WCLS, the analysis can focus solely on the learning outcomes to gauge how scores change when Bloom weighting is included. Figure 5 shows the change from UCLS to WCLS calculations for each KU once weights are applied. The KU SCA shows the smallest variation, with an increase of just 1, while Penetration Testing (PTT) exhibits the largest change, jumping by 27 and shifting in order from the 20th most burdensome KU using UCLS to the 3rd biggest lift when considering WCLS. A larger version of this graph is provided in Figure B-3.

Limitations

While the results of this analysis appear promising, there are some shortcomings. First, if there is a weakness in the administrative guidance on viewing 1 KU ≡ 1 KU, the same weakness now exists when discussing learning outcomes (LO) and topics (T), albeit perhaps to a lesser degree. For both suggested techniques, the simplification shifts to 1 LO ≡ 1 LO or 1 LO at Bloom Level X ≡ 1 LO at Bloom Level X, and 1 T ≡ 1 T. The problem with the unweighted 1 LO ≡ 1 LO is readily apparent when comparing the KU Systems Certification and Accreditation (SCA) LO #2, "Define certification and accreditation" (Becker, et al, 2024, p. 108), with the KU Embedded Systems (EBS) LO #5, "Design, develop and prototype embedded system solutions that address specific real-world problems, integrating hardware and software components effectively" (p. 63). This problem is partly mitigated by weighting learning outcomes by Bloom level, but it remains a problem nonetheless.

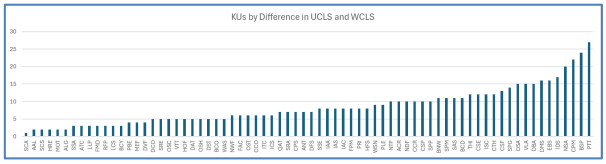

Figure 5: WCLS KUs Objective Difference from UCLS

An insidious limitation of WCLS is that LO weighting by Bloom level is ordinal, not interval. This means that it is inaccurate to consider an LO mapped to Bloom level 2 to be twice as difficult or burdensome as an LO mapped to Bloom level 1; and, by extension, we do not suggest that *creating* (Bloom level 6) is 6x more difficult than *remembering* (Bloom level 1). This can pose challenges when deriving insights from the rankings. It is crucial to remember that while the WCLS can be used to rank KUs, arithmetic operations should not be performed with it. Unfortunately, while both SCA (WCLS=8) and Supply Chain Security (SCS) (WCLS=10) are each less burdensome than Virtualization Technologies (VTT) (WCLS=18), this does not mean that SCA + SCS ≡ VTT.

When it comes to weighting, the key lies in effectively mapping verbs to Bloom's levels. That is why it is crucial for the chosen verbs to accurately represent those levels. This not only enhances clarity but also ensures that the assessments are meaningful and aligned with learning objectives.

Implications and Recommendations

This investigation suggests that the academic

burden of KUs, as indicated by unweighted and weighted curricular load scores, differs sufficiently that assuming 1 KU ≡ 1 KU is a bit tenuous. From this premise, a few suggestions are recommended:

1-that the NCAE-C program office consider forming a small task force to consider the feasibility and potential value of calculating the curricular load for KUs.

2-that the NCAE-C program office provide a Revised Bloom's Taxonomy chart of non-duplicated verbs as an appendix to the KU document for any verbs used to create KU learning outcomes – perhaps with the next document iteration.

3-a reduction in the number of verbs used across all KUs with a focus on picking verbs that have wide agreement for mapping to a particular Bloom level. In the absence of a universal, authoritative list of non-repeating verbs aligned to the revised Bloom's Taxonomy levels, it seems a good idea for significant collaborative efforts like the NCAE-C to limit the use of verbs to those that have high agreement for Bloom's level mapping.

4-that no verb be used for a single KU learning outcome. Any verb used in the KU document should be used widely.

5-that some verbs be avoided entirely to provide greater clarity of learning outcomes; e.g., leverage, "provide a summary" [summarize], contrast [see definition of compare].

6-that no learning outcome contains more than one action verb. The presence of multiple verbs, especially verbs that differ widely in Bloom category, created a problem for the WCLS method and likely causes confusion more generally.

6. CONCLUSIONS & FUTURE WORK

This paper suggests that using Knowledge Units (KUs) as an element in estimating curricular load equivalence may mask differences in the curricular burden across KUs. It is discussed that a calculation based on the underlying KU components (learning outcomes and topics) may provide greater insight and prove more useful. Two methods were described and discussed for quantifying KU curricular load: unweighted curricular load scores (UCLS) and weighted curricular load scores (WCLS). By calculating UCLS and WCLS and by documenting a transparent coding methodology, a practical tool is introduced. Though specifically used to analyze KUs under the NCAE-C program, this tool can be used more generally for curriculum mapping, course sequencing, and equitable distribution of academic content.

These measures can be adapted by any standards-aligned curriculum with defined outcomes and topics, and assist the academic units with program design and review.

Future Work

While the current research explores a measure of curricular load, future work should extend this concept to examine its pedagogical consequences. For example, higher UCLS or WCLS scores may necessitate longer instructional coverage or more complex assignments (e.g., labs versus quizzes), which directly affect course sequencing, credit hour allocation, and student workload. Investigating these connections could lead to a more equitable distribution of content across programs, improving both instructional planning and the student learning experience.

Future research should also explore empirical relationships between curricular load scores and student-centered outcomes. High-load KUs may correlate with performance gaps if faculty do not provide appropriate scaffolding or support. Building on computing education research showing that cognitive complexity strongly shapes student achievement and persistence, studies could examine how UCLS and WCLS align with grades, retention, and standardized assessment results. By connecting curricular load to instructional practices and performance data, this framework could evolve into a practical tool not only for accreditation and program design but also for advancing equity and student success in cybersecurity education.

7. REFERENCES

Application Process and Adjudication Rubric Cyber Defense Working Group. (2024, July). National Centers of Academic Excellence in Cybersecurity NCAE-C 2024 Designation Requirements and Application Process. 20240716_CAE2024_ CAE-CD_Designation_ Requirements_Ver1.19. https://dl.dod.cyber.mil/wp-content/uploads/cae/pdf/unclass-cae-cd_designation_requirements.pdf

Bamkole, A., Geissler, M., Koumadi, K., Servin, C., Tang, C., & Tucker, C. S. (2023). Bloom's for Computing: Enhancing Bloom's Revised Taxonomy with Verbs for Computing

Disciplines. Association for Computing Machinery, New York, NY. https://ccecc.acm.org/files/publications/Blooms-for-Computing-20240814.pdf

Becker, A., Blum, Z., Burgin, K. Carlin, A., Chu, B., Cranford-Wesley, D., Frank, S., Ghosh, T., Hamman, S., Joyce, R., Keller S., Kohnke, A., Levy, Y., Liu, X., Manikas, T., McBride, S., Mierzwa, S., Miller, S., Magaishi, M., … Zanella, G. (2024, December 9). National Centers of Academic Excellence in Cybersecurity (NCAE-C) – Cyber Defense (CAE-CD) Knowledge Units (KUs). https://dl.dod.cyber.mil/wp-content/uploads/cae/pdf/unclass-cae-cd_ku.pdf

Bloom, B.S. (Ed.), Engelhart, M.D., Furst, E.J., Hill,W.H., & Krathwohl, D.R. (1956). Taxonomy of educational objectives: The classification of educational goals. Handbook 1: Cognitive domain. New York: David McKay.

Clark, U., Stoker, G., Vetter, R., (2020). Looking Ahead to CAE-CD Program Changes. Information Systems Education Journal18(1) pp 29-39. https://isedj.org/2020-18/n1/ISEDJv18n1p29.pdf

Das, S., Das Mandal, S. K., & Basu, A. (2022). Classification of action verbs of Bloom's taxonomy cognitive domain: An empirical study. Journal of Education, 202(4), 554-566. https://doi.org/10.1177/00220574211002199

ElJishi, Z., Abdel-Hameed, F. S., Khuddro, A., & Zayed, S. Y. (2024). Translating bloom's taxonomy action verb list into Arabic for teacher preparation programs: Challenges/Problems and solutions. International Journal of Education and Literacy Studies, 12(1), 295-303. https://journals.aiac.org.au/index.php/IJELS/article/viewFile/8032/5254

Krathwohl, D. R. (2002). A Revision of Bloom's Taxonomy: An Overview. Theory Into Practice, 41(4), 212–218. https://doi.org/10.1207/s15430421tip4104_2

Newton, P. M., Da Silva, A., & Peters, L. G. (2020, July). A pragmatic master list of action verbs for bloom's taxonomy. In Frontiers in Education (Vol. 5, p. 107). Frontiers Media SA. https://doi.org/10.3389/feduc.2020.00107

Stanny, C. J. (2016). Reevaluating Bloom's Taxonomy: What measurable verbs can and cannot say about student learning. Education Sciences, 6(4), 37. https://doi.org/10.3390/educsci6040037

Zhou, M., & Brown, D. (2015). Educational learning theories (2nd ed.). Education Open Textbooks. https://oer.galileo.usg.edu/education-textbooks/1

Appendix A

Knowledge	*f*	Understand	*f*	Apply	*f*	Analyze	*f*	Evaluate	*f*	Create	*f*
cite	17	classify	18	act	19	analyze	24	appraise	22	arrange	22
define	21	compare	11	apply	22	appraise	11	argue	12	assemble	14
describe	14	convert	13	calculate	10	categorize	19	assess	17	combine	14
identify	20	defend	12	choose	11	classify	10	choose	10	compose	19
label	21	describe	22	compute	10	compare	24	compare	18	construct	29
list	27	discuss	21	construct	13	contrast	19	conclude	13	create	19
locate	10	distinguish	12	demonstrate	20	criticize	11	criticize	11	design	24
match	14	estimate	11	dramatize	16	diagram	12	critique	14	develop	18
memorize	10	explain	28	employ	16	differentiate	20	defend	15	devise	13
name	22	express	17	illustrate	18	discriminate	11	estimate	15	formulate	18
outline	11	extend	11	interpret	15	distinguish	21	evaluate	16	generate	11
recall	24	generalize	11	manipulate	10	divide	12	judge	25	invent	10
recite	12	identify	14	modify	12	examine	18	manage	15	modify	10
recognize	14	infer	15	operate	17	infer	14	prepare	12	organize	21
record	13	interpret	17	practice	15	outline	10	rearrange	19	plan	21
relate	11	locate	10	prepare	11	point out	12	reconcile	12	prepare	12
repeat	20	paraphrase	22	produce	13	question	12	set up	15	produce	13
reproduce	11	predict	12	relate	12	relate	17	synthesize	16	rate	21
select	16	recognize	11	schedule	11	select	12			revise	12
state	23	report	10	show	13	separate	10			write	17
		restate	15	sketch	17	subdivide	10				
		review	15	solve	19	test	14				
		rewrite	12	use	25						
		summarize	20								
		translate	21								

Figure A-1: Stanny's Table 2 of 128 verbs; 104 unique, 18 duplicates (e.g., describe under Knowledge & Understand), 3 triplicates (e.g., relate under Knowledge, Apply, & Analyze). The f score indicates the number of lists out of 30 (2016, p. 7).

Evaluation	Rate, evaluate, assess, judge, justify
Synthesis	Create, compose, argue, design, plan, support, revise, formulate
Analysis	Analyze, question, differentiate, experiment, examine, test, categorize, distinguish, calculate, contrast, outline, infer, discriminate, compare
Application	Operate, apply, use, demonstrate, solve, produce, prepare, choose
Comprehension	Translate, paraphrase, discuss, report, locate, generalize, explain, classify, summarize
Knowledge	List, define, recall, state, label, repeat, name
Avoid	*appreciate, know, familiar, aware, understand, select, explain, relate, arrange, choose*

Figure A-2: Newton et al.'s Table 1 of 51 unique verbs compiled from 47 lists. Verbs appeared in more than half of the 47 lists and were in the same Bloom level for more than half of the lists in which they were included (2020, p. 4).

Knowledge	f	Comprehension	f	Application	f	Analysis	f	Evaluation	f	Create	f
Cite	17	convert	13	act	19	Analyze	24	argue	12	Arrange	22
Define	21	discuss	21	apply	22	categorize	19	assess	17	assemble	14
Label	21	explain	28	calculate	10	Contrast	19	conclude	13	combine	14
List	27	express	17	compute	10	Diagram	12	critique	14	compose	19
Match	14	extend	17	demonstrate	20	differentiate	20	evaluate	16	create	19
memorize	10	generalize	11	dramatize	16	discriminate	11	judge	25	design	24
Name	22	paraphrase	22	employ	16	Divide	12	manage	15	develop	18
Recall	24	predict	12	illustrate	18	Examine	18	rearrange	19	devise	13
Recite	12	report	10	manipulate	10	point out	12	reconcile	12	formulate	18
Record	13	restate	15	operate	17	Question	12	set up	15	generate	11
Repeat	20	review	15	practice	15	Separate	10	synthesize	16	invent	10
reproduce	11	rewrite	12	schedule	11	subdivide	10			organize	21
State	23	summarize	20	show	13	Test	14			plan	21
		translate	21	sketch	17					rate	21
				solve	19					revise	12
				use	25					write	17

Figure A-3: Das et al.'s Table 5 of 84 unique verbs derived from Stanny's Table 2 with repeated verbs removed (2022, p. 561).

Remembering	Understanding	Applying		Analyzing	Evaluating	Creating
Define	**Annotate**	Apply	Investigate	Analyze	**Adapt**	Assemble
Duplicate	Classify	**Backup**	**Iterate**	**Articulate**	**Administer**	**Collaborate**
Enumerate	**Comment**	**Build**	Manipulate	Attribute	Appraise	**Compose**
Find	Convert	Calculate	**Map**	**Automate**	Argue	Construct
Identify	Demonstrate	Carry out	Measure	Categorize	Assess	Create
Label	Describe	**Code**	Modify	Compare	Choose	Design
List	Differentiate	**Compile**	Operate	**Contextualize**	Critique	Develop
Locate	Discuss	**Compute**	Perform	Contrast	Debate	Devise
Memorize	Exemplify	**Configure**	Produce	**Correlate**	**Debug**	Formulate
Name	Explain	**Connect**	Provision	Decompose	**Decide**	**Generate**
Recall	Infer	**Decrypt**	Randomize	Deconstruct	Defend	Hypothesize
Recognize	Interpret	**Deploy**	Recover	Deduce	Estimate	Invent
Reference	Paraphrase	Diagram	Restore	**Detect**	Evaluate	Make
Retrieve	Report	**Document**	Schedule	Discriminate	Judge	Plan
Select	Summarize	Edit	Solve	Distinguish	Justify	**Program**
State	Translate	**Encrypt**	Store	Examine	**Optimize**	**Script**
		Execute	Train	**Generalize**	**Prioritize**	**Secure**
		Graph	Use	Integrate	**Prove**	**Visualize**
		Illustrate	Virtualize	**Model**	Support	
		Implement	Write	**Monitor**	Test	
		Install		Organize	**Validate**	
				Outline	Value	
				Predict	Verify	
				Simulate		
				Structure		
				Trace		
				Translate		
				Update		

Figure A-4: Bamkole et al.'s Bloom's for Computing list of 142 unique verbs, 56 of which are the new compute-related verbs (2023, p. 28).

	Action verbs
Remember تذكر	Find, cite, locate, recall, highlight, retrieve, search, define, describe, label, list, match, name, reproduce, state
Understand افهم	Annotate, outline, compare, discuss, convert, explain, extend, generalize, exemplify (give an example), paraphrase, predict, summarize, translate, research, review, restate
Apply طبّق	Apply, articulate, calculate, choose, complete, execute, dramatize, practice, share, change, illustrate, operate, teach, examine, classify, compute, demonstrate, discover, manipulate, prepare, produce, relate, show, solve, use
Analyze حلّل	Analyze, categorize, deduce, edit, investigate, reverse, select, separate, engineer, examine, establish, break down, conclude, diagram, deconstruct, differentiate, discriminate, distinguish, correlate, contrast
Evaluate قيّم	Argue, assess, collaborate, critique, debate, evaluate, hypothesize, judge, moderate, recommend, reflect, test, verify, prioritize, rate, inspect, decide, measure. appraise, conclude, criticize, defend, discriminates, justify, support
Create أبدع	Integrate, intervene, model, negotiate, plan, progress, rearrange, formulate, construct, reinforce, revise, structure, substitute, validate, assemble, develop, draft, invent, produce, propose, publish, repurpose, upload, write, synthesize, categorize, combine, compile, compose, create, devise, design, generate, organize, reconstruct, reorganize, rewrite, tell, identify

Figure A-5: ElJishi et al.'s Table 1 140-verb list with four duplicates across levels – categorize, conclude, examine, and produce (2024, p. 298).

Appendix B

List of Unique KU Verbs as Placed in Revised Bloom's Taxonomy Levels					
	ElJishi (2024)	**ACM (2023)**	**Das (2022)**	**Newton (2020)**	**Authors**
Remember					
define	x	x	x	x	
identify		x			
list	x	x	x	x	
recall	x	x	x	x	
recognize		x			
select		x			
Understand					
annotate	x	x			
communicate					x
describe					x
discuss	x	x	x	x	
explain	x	x	x	x	
explore					x
review	x		x		
understand					x
Apply					
apply	x	x	x	x	
assist					x
compute	x	x	x		
conduct					x
configure		x			
demonstrate	x		x	x	
deploy		x			
document		x			
draw					x
execute	x	x			
handle					x
harden					x
illustrate	x	x	x		
implement		x			
install		x			
leverage					x
map		x			
mitigate					x
perform		x			
produce		x		x	

protect					x
provide					x
quantify					x
use	x	x	x	x	
utilize					x
	ElJishi (2024)	**ACM (2023)**	**Das (2022)**	**Newton (2020)**	**Authors**
Analyze					
analyze	x	x	x	x	
articulate		x			
categorize		x	x	x	
characterize					x
compare		x		x	
contrast	x	x	x	x	
decipher					x
detect		x			
diagram	x		x		
differentiate	x		x	x	
examine		x	x	x	
monitor		x			
outline		x		x	
resolve					x
Evaluate					
assess	x	x	x	x	
determine					x
evaluate	x	x	x	x	
rate	x			x	
recommend	x				
set up			x		
suggest					x
test	x	x			
Create					
create	x	x	x	x	
design	x	x	x	x	
develop	x	x	x		
devise	x	x	x		
organize	x		x		
plan	x	x	x	x	
propose	x				
prototype					x
write	x		x		

Table B-1: reference for why verbs were placed in Bloom category

List of KUs Ordered by Unweighted Curricular Load Score (UCLS) UCLS = # of Learning Outcomes (LO) + # of Topics (T)							
KU	**LO**	**T**	**UCLS**	**KU**	**LO**	**T**	**UCLS**
Hardware/Firmware Security (HFS)	5	41	46	Introduction to Theory of Computation (ITC)	3	15	18
Intrusion Detection/Prevention Systems (IDS)	7	31	38	Fraud Prevention & Management (FPM)	7	10	17
Cybersecurity Fundamentals (CSF)	10	25	35	Threat Intelligence (THI)	7	10	17
Secure Programming Practices (SPP)	5	28	33	Network Technology & Protocols (NTP)	6	11	17
Policy, Legal, Ethics, and Compliance (PLE)	6	26	32	Operating Systems Hardening (OSH)	3	14	17
Data Administration (DBA)	5	27	32	Operating Systems Theory (OST)	2	15	17
Business Continuity and Disaster Recovery (BCD)	5	25	30	Cyber-Physical Systems (CPS)	3	13	16
Industrial Control Systems (ICS)	6	23	29	Databases (DAT)	3	13	16
Digital Forensics (DFS)	5	24	29	IA Compliance (IAC)	2	14	16
Cyber Threats (CTH)	4	25	29	Security Risk Analysis (SRA)	5	10	15
IT Systems Components (ISC)	7	21	28	IA Standards (IAS)	5	10	15
Cybersecurity Principles (CSP)	5	23	28	Host Forensics (HOF)	4	11	15
Software Assurance (SAS)	5	21	26	Operating Systems Concepts (OSC)	4	9	13
Cyber Crime (CCR)	3	23	26	Forensic Accounting (FAC)	4	9	13
Embedded Systems (EBS)	5	19	24	Radio Frequency Principles (RFP)	3	10	13
Basic Cryptography (BCY)	4	20	24	Virtualization Technologies (VTT)	3	10	13
Security Program Management (SPM)	4	20	24	Device Forensics (DVF)	3	9	12
Advanced Network Technology & Protocols (ANT)	3	21	24	Network Forensics (NWF)	3	9	12
Penetration Testing (PTT)	8	15	23	Media Forensics (MEF)	3	8	11
Basic Scripting and Programming (BSP)	6	17	23	Formal Methods (FMD)	2	9	11
Cybersecurity Planning and Management (CPM)	6	17	23	IA Architectures (IAA)	2	9	11
Database Management Systems (DMS)	6	17	23	Algorithms (ALG)	1	10	11
Systems Programming (SPG)	4	19	23	Low Level Programming (LLP)	2	8	10
Basic Cyber Operations (BCO)	5	17	22	Mobile Technologies (MOT)	2	8	10
Network Defense (NDF)	4	18	22	Systems Security Engineering (SSE)	2	8	10
Privacy (PRI)	4	18	22	Hardware Reverse Engineering (HRE)	1	9	10
Web Application Security (WAS)	3	19	22	Pre-OS Boot Environment (PBE)	3	6	9
Operating Systems Administration (OSA)	7	14	21	Analog Telecommunications (ATC)	2	7	9
Cybersecurity Ethics (CSE)	6	15	21	Software Reverse Engineering (SRE)	2	7	9
Vulnerability Analysis (VLA)	5	16	21	Digital Communications (DCO)	3	5	8
Wireless Sensor Networks (WSN)	3	18	21	QA/Functional Testing (QAT)	2	6	8
Network Security Administration (NSA)	7	13	20	Supply Chain Security (SCS)	2	6	8
Life-Cycle Security (LCS)	3	17	20	Advanced Algorithms (AAL)	1	7	8
Data Structures (DST)	4	15	19	Software Security Analysis (SSA)	2	5	7
Basic Networking (BNW)	6	12	18	Systems Certification & Accreditation (SCA)	2	5	7
Advanced Cryptography (ACR)	4	14	18	Independent/Directed Study/Research (IDR)			N/A
Cloud Computing (CCO)	4	14	18				

Table B-2: List of all 73 KUs ordered by UCLS

List of KUs Ordered by Weighted Curricular Load Score (WCLS)							
KU	**LO**	**T**	**WCLS**	**KU**	**LO**	**T**	**WCLS**
Intrusion Detection/Prevention Systems (IDS)	7	31	55	Cloud Computing (CCO)	4	14	24
Hardware/Firmware Security (HFS)	5	41	54	IA Compliance (IAC)	2	14	24
Penetration Testing (PTT)	8	15	50	Data Structures (DST)	4	15	24
Cybersecurity Fundamentals (CSF)	10	25	48	Introduction to Theory of Computation (ITC)	3	15	24
Basic Scripting and Programming (BSP)	6	17	47	IA Standards (IAS)	5	10	23
Data Administration (DBA)	5	27	47	Cyber-Physical Systems (CPS)	3	13	23
Cybersecurity Planning and Management (CPM)	6	17	45	Operating Systems Theory (OST)	2	15	23
Secure Programming Practices (SPP)	5	28	43	Life-Cycle Security (LCS)	3	17	23
Business Continuity and Disaster Recovery (BCD)	5	25	41	Security Risk Analysis (SRA)	5	10	22
Cyber Threats (CTH)	4	25	41	Operating Systems Hardening (OSH)	3	14	22
Policy, Legal, Ethics, and Compliance (PLE)	6	26	41	Databases (DAT)	3	13	21
Network Security Administration (NSA)	7	13	40	Host Forensics (HOF)	4	11	20
Embedded Systems (EBS)	5	19	40	Forensic Accounting (FAC)	4	9	19
IT Systems Components (ISC)	7	21	40	IA Architectures (IAA)	2	9	19
Database Management Systems (DMS)	6	17	39	Systems Security Engineering (SSE)	2	8	18
Cybersecurity Principles (CSP)	5	23	38	Operating Systems Concepts (OSC)	4	9	18
Systems Programming (SPG)	4	19	37	Network Forensics (NWF)	3	9	18
Software Assurance (SAS)	5	21	37	Virtualization Technologies (VTT)	3	10	18
Operating Systems Administration (OSA)	7	14	36	Device Forensics (DVF)	3	9	16
Vulnerability Analysis (VLA)	5	16	36	Radio Frequency Principles (RFP)	3	10	16
Cyber Crime (CCR)	3	23	36	QA/Functional Testing (QAT)	2	6	15
Digital Forensics (DFS)	5	24	36	Media Forensics (MEF)	3	8	15
Security Program Management (SPM)	4	20	35	Software Reverse Engineering (SRE)	2	7	14
Industrial Control Systems (ICS)	6	23	35	Formal Methods (FMD)	2	9	14
Cybersecurity Ethics (CSE)	6	15	33	Digital Communications (DCO)	3	5	13
Network Defense (NDF)	4	18	32	Pre-OS Boot Environment (PBE)	3	6	13
Advanced Network Technology & Protocols (ANT)	3	21	31	Low Level Programming (LLP)	2	8	13
Privacy (PRI)	4	18	30	Algorithms (ALG)	1	10	13
Wireless Sensor Networks (WSN)	3	18	30	Analog Telecommunications (ATC)	2	7	12
Threat Intelligence (THI)	7	10	29	Mobile Technologies (MOT)	2	8	12
Basic Networking (BNW)	6	12	29	Hardware Reverse Engineering (HRE)	1	9	12
Advanced Cryptography (ACR)	4	14	28	Software Security Analysis (SSA)	2	5	10
Network Technology & Protocols (NTP)	6	11	27	Supply Chain Security (SCS)	2	6	10
Basic Cyber Operations (BCO)	5	17	27	Advanced Algorithms (AAL)	1	7	10
Web Application Security (WAS)	3	19	27	Systems Certification & Accreditation (SCA)	2	5	8
Basic Cryptography (BCY)	4	20	27	Independent/Directed Study/Research (IDR)			N/A
Fraud Prevention & Management (FPM)	7	10	25				

Table B-3: List of all 73 KUs ordered by WCLS

List of KUs and Learning Objective Verbs Used for Weighting Ordered by Verb Weight			
KU	**LO**	**Verb weight**	**LO Verbs Used for Weighting**
Penetration Testing (PTT)	8	35	plan, analyze, discuss, describe, create, devise, assess, compare
Basic Scripting and Programming (BSP)	6	30	write, write, write, write, implement, demonstrate
Cybersecurity Planning and Management (CPM)	6	28	examine, develop, develop, outline, discuss, develop
Network Security Administration (NSA)	7	27	recommend, recommend, protect, monitor, assist, evaluate, discuss
Intrusion Detection/Prevention Systems (IDS)	7	24	detect, apply, apply, leverage, apply, test, apply
Cybersecurity Fundamentals (CSF)	10	23	define, describe, describe, describe, evaluate, describe, describe, apply, describe, discuss
Database Management Systems (DMS)	6	22	compare, describe, apply, apply, outline, design
Operating Systems Administration (OSA)	7	22	set up, configure, configure, perform, install, review, configure
Embedded Systems (EBS)	5	21	describe, explain, develop, evaluate, design
Data Administration (DBA)	5	20	draw, evaluate, examine, compare, outline
Vulnerability Analysis (VLA)	5	20	apply, create, apply, propose, explain
IT Systems Components (ISC)	7	19	differentiate, characterize, describe, understand, understand, describe, apply
Threat Intelligence (THI)	7	19	identify, perform, apply, demonstrate, demonstrate, apply, apply
Systems Programming (SPG)	4	18	develop, apply, implement, develop
Cybersecurity Ethics (CSE)	6	18	explain, examine, describe, identify, examine, assess
Basic Networking (BNW)	6	17	describe, apply, apply, apply, examine, describe
Cyber Threats (CTH)	4	16	compare, rate, evaluate, explain
Business Continuity and Disaster Recovery (BCD)	5	16	identify, explain, implement, suggest, evaluate
Software Assurance (SAS)	5	16	apply, describe, create, apply, explain
Network Technology & Protocols (NTP)	6	16	demonstrate, demonstrate, describe, mitigate, demonstrate, explain
Security Program Management (SPM)	4	15	apply, apply, assess, articulate
Cybersecurity Principles (CSP)	5	15	differentiate, describe, analyze, apply, understand
Secure Programming Practices (SPP)	5	15	produce, describe, understand, differentiate, examine
Policy, Legal, Ethics, and Compliance (PLE)	6	15	describe, describe, differentiate, explain, explain, apply
Fraud Prevention & Management (FPM)	7	15	describe, describe, analyze, describe, describe, describe, recognize
Network Defense (NDF)	4	14	describe, explain, evaluate, evaluate
Advanced Cryptography (ACR)	4	14	explain, evaluate, explain, evaluate
Cyber Crime (CCR)	3	13	examine, evaluate, examine
Hardware/Firmware Security (HFS)	5	13	outline, use, describe, describe, discuss
IA Standards (IAS)	5	13	compare, map, describe, describe, describe
Wireless Sensor Networks (WSN)	3	12	diagram, describe, propose
Privacy (PRI)	4	12	examine, explore, describe, compare
Security Risk Analysis (SRA)	5	12	describe, describe, evaluate, identify, annotate
Digital Forensics (DFS)	5	12	discuss, describe, describe, use, perform
Industrial Control Systems (ICS)	6	12	identify, describe, describe, apply, explain, explain
IA Architectures (IAA)	2	10	examine, design
IA Compliance (IAC)	2	10	compare, plan

Systems Security Engineering (SSE)	2	10	determine, determine
Advanced Network Technology & Protocols (ANT)	3	10	describe, describe, develop
Cyber-Physical Systems (CPS)	3	10	describe, implement, evaluate
Cloud Computing (CCO)	4	10	compare, list, explain, apply
Forensic Accounting (FAC)	4	10	describe, implement, describe, compute
Basic Cyber Operations (BCO)	5	10	describe, describe, identify, describe, use
QA/Functional Testing (QAT)	2	9	develop, perform
Introduction to Theory of Computation (ITC)	3	9	describe, differentiate, quantify
Network Forensics (NWF)	3	9	describe, analyze, use
Operating Systems Concepts (OSC)	4	9	describe, describe, describe, install
Data Structures (DST)	4	9	list, discuss, utilize, implement
Host Forensics (HOF)	4	9	discuss, describe, describe, perform
Operating Systems Theory (OST)	2	8	understand, design
Databases (DAT)	3	8	describe, outline, describe
Digital Communications (DCO)	3	8	describe, describe, compare
Operating Systems Hardening (OSH)	3	8	describe, install, leverage
Virtualization Technologies (VTT)	3	8	describe, compare, discuss
Web Application Security (WAS)	3	8	examine, describe, explain
Software Reverse Engineering (SRE)	2	7	apply, analyze
Device Forensics (DVF)	3	7	describe, perform, explain
Media Forensics (MEF)	3	7	describe, apply, explain
Pre-OS Boot Environment (PBE)	3	7	describe, describe, demonstrate
Basic Cryptography (BCY)	4	7	identify, describe, describe, describe
Life-Cycle Security (LCS)	3	6	describe, describe, describe
Radio Frequency Principles (RFP)	3	6	understand, understand, discuss
Analog Telecommunications (ATC)	2	5	illustrate, understand
Formal Methods (FMD)	2	5	apply, describe
Low Level Programming (LLP)	2	5	apply, explain
Software Security Analysis (SSA)	2	5	describe, apply
Mobile Technologies (MOT)	2	4	understand, describe
Supply Chain Security (SCS)	2	4	describe, describe
Advanced Algorithms (AAL)	1	3	implement
Algorithms (ALG)	1	3	implement
Hardware Reverse Engineering (HRE)	1	3	perform
Systems Certification & Accreditation (SCA)	2	3	describe, define
Independent/Directed Study/Research (IDR)		N/A	

Table B-4: verbs and verb weighting used for each KU for the WCLS calculation; e.g., the KU

List of KU Learning Outcomes (LO) with Multiple Verbs; with Verbs Unused for Weighting Identified E.g., CSF LO #5 has verbs describe & evaluate; from Table B-1, describe is Bloom Level 2, evaluate is Bloom Level 5, so evaluate is used when calculating the WCLS			
KU	**LO#**	**All Verbs**	**Verbs Unused for Weighting**
Cybersecurity Fundamentals (CSF)	5	describe, evaluate	describe
Cybersecurity Principles (CSP)	1	differentiate, discuss	discuss
Cybersecurity Principles (CSP)	3	analyze, identify	identify
Cybersecurity Principles (CSP)	4	identify, apply	identify
IT Systems Components (ISC)	1	differentiate, diagram	differentiate
Basic Networking (BNW)	1	describe, explain	explain
Basic Networking (BNW)	3	apply, demonstrate	demonstrate
Basic Networking (BNW)	4	apply, demonstrate	demonstrate
Basic Networking (BNW)	5	perform, examine	perform
Basic Scripting and Programming (BSP)	1	write, execute	execute
Basic Scripting and Programming (BSP)	2	write, execute	execute
Basic Scripting and Programming (BSP)	3	write, execute	execute
Basic Scripting and Programming (BSP)	4	write, execute	execute
Network Defense (NDF)	1	describe, discuss	discuss
Network Defense (NDF)	2	explain, discuss	explain
Network Defense (NDF)	3	analyze, evaluate	analyze
Operating Systems Concepts (OSC)	1	describe, discuss	describe
Operating Systems Concepts (OSC)	2	describe, discuss	describe
Operating Systems Concepts (OSC)	3	identify, describe	identify
Operating Systems Concepts (OSC)	4	install, configure, harden	configure, harden
Cyber Threats (CTH)	1	identify, compare, contrast	identify, compare
Cyber Threats (CTH)	2	communicate, rate, describe	communicate, describe
Cyber Threats (CTH)	4	explain, discuss	explain
Cybersecurity Planning and Management (CPM)	1	examine, describe	describe
Cybersecurity Planning and Management (CPM)	4	outline, explain	explain
Policy, Legal, Ethics, and Compliance (PLE)	1	identify, recall, describe	identify, recall
Policy, Legal, Ethics, and Compliance (PLE)	3	describe, differentiate	describe
Security Risk Analysis (SRA)	1	describe, explain	describe
Security Risk Analysis (SRA)	3	evaluate, categorize, recommend	evaluate, categorize
Security Risk Analysis (SRA)	4	identify, select	select
Security Risk Analysis (SRA)	5	annotate, apply	annotate
Advanced Algorithms (AAL)	1	understand, implement	understand
Advanced Cryptography (ACR)	4	evaluate, explain	explain
Advanced Network Technology & Protocols (ANT)	1	identify, describe	identify
Advanced Network Technology & Protocols (ANT)	2	describe, discuss	discuss
Algorithms (ALG)	1	understand, implement	understand
Analog Telecommunications (ATC)	1	describe, illustrate	describe

Analog Telecommunications (ATC)	2	understand, describe	understand
Business Continuity and Disaster Recovery (BCD)	2	explain, describe	explain
Business Continuity and Disaster Recovery (BCD)	4	analyze, suggest	analyze
Business Continuity and Disaster Recovery (BCD)	5	evaluate, recommend	recommend
Basic Cyber Operations (BCO)	2	list, describe	list
Cloud Computing (CCO)	4	describe, apply	describe
Cyber-Physical Systems (CPS)	3	analyze, evaluate	analyze
Cybersecurity Ethics (CSE)	4	identify, recall	recall
Cybersecurity Ethics (CSE)	5	examine, differentiate	differentiate
Data Administration (DBA)	1	draw, describe	describe
Data Administration (DBA)	2	define, evaluate	define
Data Administration (DBA)	4	compare, contrast	contrast
Database Management Systems (DMS)	1	compare, contrast	contrast
Database Management Systems (DMS)	4	describe, apply	describe
Database Management Systems (DMS)	6	design, deploy	deploy
Databases (DAT)	3	identify, describe	identify
Device Forensics (DVF)	2	perform, handle, understand	handle, understand
Digital Communications (DCO)	3	compare, contrast, describe	contrast, describe
Digital Forensics (DFS)	2	identify, describe	identify
Embedded Systems (EBS)	1	identify, describe	identify
Embedded Systems (EBS)	3	develop, implement	implement
Embedded Systems (EBS)	5	design, develop, prototype	develop, prototype
Forensic Accounting (FAC)	2	describe, implement	implement
Hardware/Firmware Security (HFS)	2	explain, use	explain
Host Forensics (HOF)	4	perform, provide	provide
IA Architectures (IAA)	1	examine, identify	identify
IA Compliance (IAC)	1	compare, contrast	contrast
IA Compliance (IAC)	2	plan, conduct	conduct
IA Standards (IAS)	1	compare, contrast	contrast
IA Standards (IAS)	5	list, describe	list
Industrial Control Systems (ICS)	1	identify, recall	identify
Introduction to Theory of Computation (ITC)	3	describe, quantify	describe
Intrusion Detection/Prevention Systems (IDS)	1	detect, identify, resolve, document	identify, resolve, document
Intrusion Detection/Prevention Systems (IDS)	6	deploy, test	deploy
Life-Cycle Security (LCS)	2	list, describe, explain	list, describe
Life-Cycle Security (LCS)	3	list, describe	list
Mobile Technologies (MOT)	1	understand, explain	understand
Network Forensics (NWF)	2	analyze, decipher, identify, provide	decipher, identify, provide
Network Security Administration (NSA)	1	analyze, recommend	analyze
Network Security Administration (NSA)	6	evaluate, perform	perform

Network Technology & Protocols (NTP)	3	identify, describe	identify
Network Technology & Protocols (NTP)	4	identify, mitigate	identify
Operating Systems Theory (OST)	2	understand, design, implement	understand, implement
Penetration Testing (PTT)	1	plan, organize, perform	organize, perform
Penetration Testing (PTT)	7	assess, determine	determine
Penetration Testing (PTT)	8	compare, contrast	contrast
Privacy (PRI)	4	compare, contrast	contrast
Radio Frequency Principles (RFP)	1	understand, identify	identify
Radio Frequency Principles (RFP)	2	understand, identify	identify
Systems Programming (SPG)	2	outline, apply	apply
Systems Security Engineering (SSE)	1	analyze, determine	analyze
Systems Security Engineering (SSE)	2	analyze, determine	analyze
Virtualization Technologies (VTT)	2	compare, contrast	contrast
Vulnerability Analysis (VLA)	2	create, apply	apply
Vulnerability Analysis (VLA)	4	propose, analyze	analyze
Wireless Sensor Networks (WSN)	1	diagram, deploy	deploy
Wireless Sensor Networks (WSN)	3	analyze, propose	analyze

Table B-5: list of KU Learning Outcomes with multiple verbs identifying which verbs were not used for the weighting when calculating WCLS

List of 70 Unique Verbs Used Across the 73 KUs Ordered by Frequency of Use (402 total verb uses)						
Verb	**# Times Used**	**% Verb Uses**		**Verb**	**# Times Used**	**% Verb Uses**
describe	91	22.6%		recall	2	0.5%
apply	32	8.0%		articulate	1	0.2%
explain	26	6.5%		assist	1	0.2%
identify	24	6.0%		categorize	1	0.2%
understand	15	3.7%		characterize	1	0.2%
evaluate	14	3.5%		communicate	1	0.2%
analyze	13	3.2%		compute	1	0.2%
discuss	13	3.2%		conduct	1	0.2%
compare	11	2.7%		decipher	1	0.2%
examine	11	2.7%		detect	1	0.2%
implement	11	2.7%		devise	1	0.2%
perform	10	2.5%		diagram	1	0.2%
demonstrate	9	2.2%		document	1	0.2%
develop	9	2.2%		draw	1	0.2%
contrast	8	2.0%		explore	1	0.2%
differentiate	7	1.7%		handle	1	0.2%
list	6	1.5%		harden	1	0.2%
outline	5	1.2%		illustrate	1	0.2%
configure	4	1.0%		map	1	0.2%
design	4	1.0%		mitigate	1	0.2%
execute	4	1.0%		monitor	1	0.2%
use	4	1.0%		organize	1	0.2%
write	4	1.0%		produce	1	0.2%
assess	3	0.7%		protect	1	0.2%
create	3	0.7%		prototype	1	0.2%
define	3	0.7%		quantify	1	0.2%
deploy	3	0.7%		rate	1	0.2%
determine	3	0.7%		recognize	1	0.2%
install	3	0.7%		resolve	1	0.2%
recommend	3	0.7%		review	1	0.2%
annotate	2	0.5%		select	1	0.2%
leverage	2	0.5%		set up	1	0.2%
plan	2	0.5%		suggest	1	0.2%
propose	2	0.5%		test	1	0.2%
provide	2	0.5%		utilize	1	0.2%

Table B-6: frequency of use across 73 KUs for all 70 unique verbs

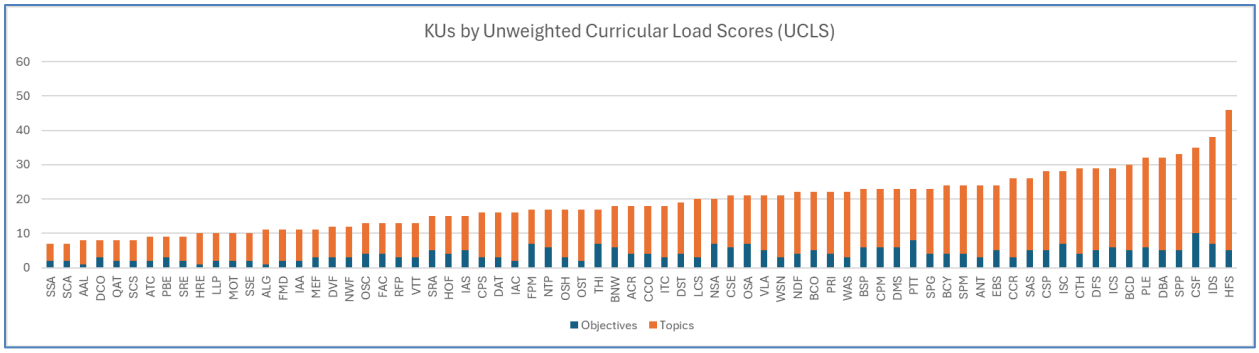

Figure B-1: KUs arranged from low to high UCLS

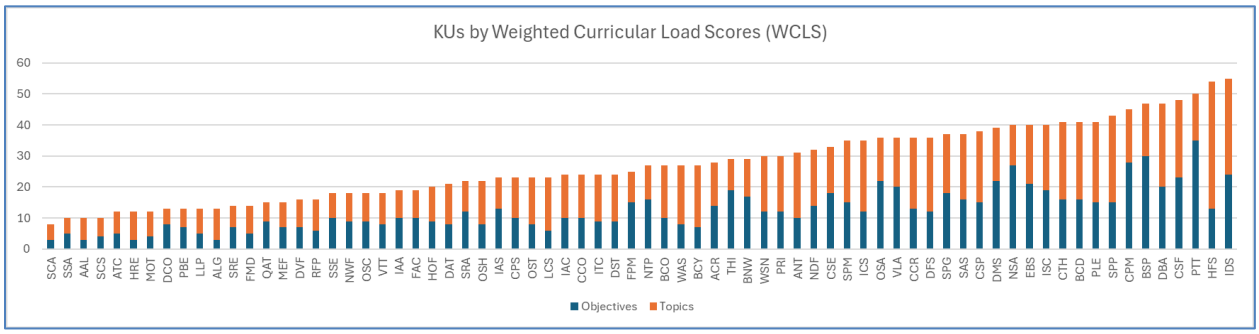

Figure B-2: KUs arranged from low to high WCLS

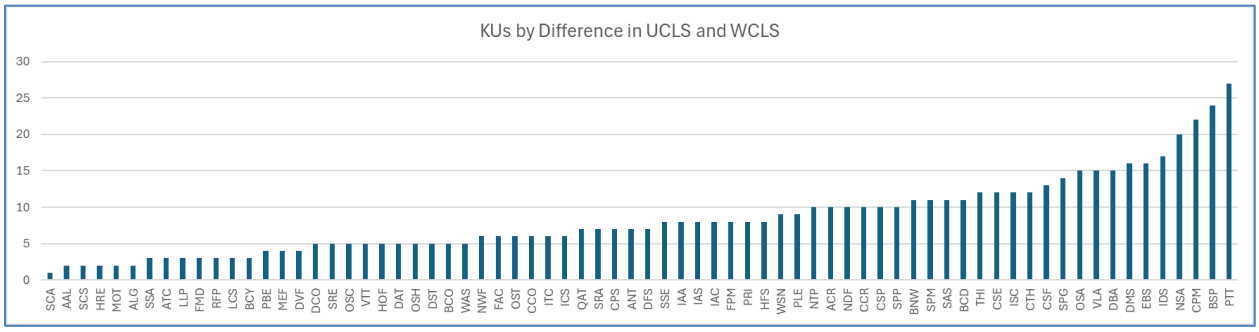

Figure B-3: KUs arranged from low to high by the change in UCLS to WCLS