Preamble

[As stated below, the Security Knowledge Area crosscuts and shares many concepts with the other CS2023 Knowledge Areas that are constantly being revised. Given these interdependencies, this version of the Security Knowledge Area should be considered as a work-in-progress version rather than a full beta release, which will need to be revised based on the other knowledge areas. Feedback is nevertheless sought on the approach being used to ensure the next version reflects the state of security within computer science in 2023.]

The world increasingly relies on computing infrastructure to support nearly every facet of modern critical infrastructure: transportation, communication, healthcare, education, energy generation and distribution, just to name a few. In recent years, with rampant attacks on and breaches of this critical infrastructure, it has become clearer that computer science graduates have an increased role in designing, implementing, and operating software systems that are secure and can keep information private.

In CS2023, the Security (SEC) Knowledge Area (KA) represents both crosscutting themes pervasive in all the other areas of CS2023, including Software Development Fundamentals, Data Management, Operating Systems, Networking and Communications, Parallel and Distributed Computing, Systems Fundamentals, and Artificial Intelligence. Consequently, a Security mindset needs to be incorporated into the overall ethos of computer science graduates so that security is inherent in all of their work product. (Also note the Security title was chosen as an umbrella term for this KA and includes concepts such as privacy, system design, and cryptography, which are included in the other KAs.)

The six crosscutting themes of cybersecurity, viewed with a computer science lens\(^1\): confidentiality, integrity, availability, risk, systems thinking, and adversarial thinking, are also relevant here. Of these, developing an adversarial thinking mindset is not typically covered in the other Computer Science Knowledge Areas (KAs), thus it is emphasized within the SEC core. Students also need to learn security concepts such as authentication, authorization, and non-repudiation. They also need to learn about system vulnerabilities and understand threats against software systems. As such, principles of protecting systems must be covered to complement system design principles covered in the SDF and SE KAs, including principles such as secure by design, privacy by design, or defense in depth. Another concept important in the SEC KA is the notion of assurance, which is an attestation that security mechanisms comply with the security policies that have been defined for data, processes, and systems. With the increased use of computing systems and data sets in modern society, the issues of privacy, especially its technical aspects not covered in the Society, Ethics and Professionalism KA.

Changes since CS 2013: The Security KA is an “updated” name for CS2013’s Information Assurance and Security (IAS) knowledge area. Since 2013, Information Assurance and Security has been rebranded as Cybersecurity, which has become a new computing discipline. Moreover, a Joint Task Force of the ACM, IEEE Computer Society, AIS and IFIP developed curricular guidelines, Cybersecurity 2017 (CSEC 2017) to reflect the new discipline. Therefore, the Security KA in CS2023 is informed by the notion of a disciplinary lens outlined in CSEC 2017 to focus on those aspects of security that are important for computer science students. CS2023’s Security KA also incorporates recent developments in computer science for securing systems and improving privacy, building on CS2013’s recognition of the pervasiveness of security in computer science. A Task Force has also been convened to develop the next decennial update of the CSEC 2017 guidelines: the focus in in CS2023 is on those aspects of security, privacy and related concepts relevant to the computer science discipline.

Core Hours

Several concepts in the Security KA are foundational elements of Computer Science curriculum for all students, as many graduates of undergraduate Computer Science programs will build software for the modern world where security is critical to the functioning of modern society. However, given the competing needs of other KAs and the limited amount of time available in most undergraduate programs, the philosophy here is to focus on security elements that students need to know that are not already required by other knowledge areas. As discussed below, the course packaging will show that most programs will be encouraged to go beyond the CS Core covered in this KA and the others listed below.

Therefore, the SEC KA CS Core Hours and KA Core Hours are shown in two parts.
1. CS Core Hours that the SEC area adds to the overall CS2023 curriculum that are not already included in the other KAs, and
2. CS Core Hours already included in the other KAs that are not meant to be duplicative, instead they are documentation that these are also important to the CS and KA Core Hours for the SEC KA. These numbers are shown in parentheses in the table below.

<table>
<thead>
<tr>
<th>Knowledge Unit</th>
<th>CS Core</th>
<th>KA Core</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foundational Security</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>AL/Algorithms and Society</td>
<td>(2)</td>
<td>1</td>
</tr>
<tr>
<td>AL/Fundamental Data Structures and Algorithms*</td>
<td>(2)</td>
<td>2</td>
</tr>
<tr>
<td>AL/Cryptography*</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>AR/All Knowledge Units</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>DM/Data and Database Security</td>
<td>(2)</td>
<td>3</td>
</tr>
<tr>
<td>Knowledge Area</td>
<td>Units</td>
<td>Total</td>
</tr>
<tr>
<td>-----------------------------------------------------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>GIT/Animation and Immersion</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>HCI/Accountability and Responsibility in Design</td>
<td>(1)</td>
<td>2</td>
</tr>
<tr>
<td>MSF/Cryptography</td>
<td>(1)</td>
<td>(4)</td>
</tr>
<tr>
<td>NC/Network Security</td>
<td>(1)</td>
<td>2</td>
</tr>
<tr>
<td>OS/Role and purpose of OS</td>
<td>(1)</td>
<td>2</td>
</tr>
<tr>
<td>OS/Protection and Safety</td>
<td>(2)</td>
<td>2</td>
</tr>
<tr>
<td>PD/Communication</td>
<td>(2)</td>
<td>2</td>
</tr>
<tr>
<td>PD/Software Engineering</td>
<td>(1)</td>
<td>0</td>
</tr>
<tr>
<td>SEP/Privacy and Civil Liberties</td>
<td>(2)</td>
<td>2</td>
</tr>
<tr>
<td>SEP/Security Policies, Laws and Computer Crimes</td>
<td>(2)</td>
<td>2</td>
</tr>
<tr>
<td>SDF/Fundamental Programming Concepts</td>
<td>(2)</td>
<td>0</td>
</tr>
<tr>
<td>SE/Product Requirements</td>
<td>(1)</td>
<td>2</td>
</tr>
<tr>
<td>SE/Software Design</td>
<td>(1)</td>
<td>3</td>
</tr>
<tr>
<td>SE/Software Construction</td>
<td>(1)</td>
<td>2</td>
</tr>
<tr>
<td>SE/Software Verification and Validation</td>
<td>(1)</td>
<td>3</td>
</tr>
<tr>
<td>SPF/All Knowledge Units</td>
<td>(2)</td>
<td>3</td>
</tr>
<tr>
<td>SF/Systems Security</td>
<td>(2)</td>
<td>3</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>4</td>
<td>45 (+ 29 included in other KAs)</td>
</tr>
</tbody>
</table>

**Knowledge Units**

**SEC/Foundational Security**

**Topics:**

- Crosscutting concepts within security: confidentiality, integrity, availability, risk, adversarial thinking, systems thinking
- Vulnerabilities, threats, and attack vectors
Authentication and authorization, and access control techniques
Concept of trust and trustworthiness
Principles of protection, e.g., least privilege, open design, fail-safe defaults, defense in depth, layered defense
Principles and practices of privacy
Tensions between security, privacy, performance, and other design goals
Legal issues
Ethical considerations

Illustrative Learning Outcomes
- Design and develop a system that is secure against a set of identified threats
- Evaluate a system for trustworthiness
- Develop a system that incorporates various principles of security and evaluate it for its resilience to attacks
- Design and develop a system designed to protect individual privacy

SEC/Other Knowledge Units

Based on the right column of the table shown above, a set of Knowledge Units will be proposed here only if there is no natural home in the other Knowledge Areas. Such knowledge units, with their topics and illustrative learning outcomes, will be included in the next version of the Security KA.

KA and SEP

Security, with its associated topics such as privacy, are both providers and consumers of the SEP KA. While SEP focuses on broader social perspectives that inform the SEC KA, the SEC KA provides the “nuts and bolts” in computer science to ensure SEP can live up to its potential.

The SEP/Privacy and Civil Liberties Knowledge Unit covers the philosophical foundations of privacy rights, along with current and future legal directions. The Security KA covers the technical side of ensuring that privacy design and engineering principles can support privacy policies and regulations, as well as reduce or eliminate any adverse impact of privacy loss. Bias in algorithms and data (and its collection) need to be addressed within the Security KA.

The SEP/Security Policies, Laws and Computer Crimes Knowledge Unit covers the issues of computer crimes, malware, criminal hacking, cyber terrorism and more, all of which need to be understood with the Security KA to ensure appropriate technical solutions and safeguards can be implemented, especially for society at large that may not be familiar with computing.

Professional Dispositions
Possible dispositions typically included in computing by CC2020 are Perseverance, Inventive, Meticulous, Self-directed (e.g., self-learner), Collaborative, Proactive, Persistent, Professional, Responsible, Adaptable, Responsive, and Accountable. Although most professional dispositions are also desirable in the Security KA, the most desirable professional dispositions for this knowledge area are:

- **Meticulous**: Careful attention must be paid to details of the real world when developing a secure system to assure that every aspect of the system is protected. This requires meticulousness on the part of the student.
- **Self-directed** (e.g., self-learner): As the adversary is always going to look into newer ways of attacking systems and breaching data, students would need to constantly learn to keep up to date with the likely problems and attacks of tomorrow so that they are prepared to defend their systems and data.
- **Collaborative**: Most security and privacy breaches require collaboration with other personnel, especially in the face of a real-time attack. Students need to be able to work and rely on their teams to prevent breaches.
- **Responsible**: As society increasingly depends on computing infrastructure and information systems, students need to show responsibility when designing, developing, deploying, and maintaining secure systems.
- **Accountable**: The protection of systems and risk mitigation requires accountability if things go wrong. Therefore, future professionals need to know that they will be held accountable for security breaches and need to do their best in terms of design and implementation to prevent such breaches from occurring.

### Math Requirements

**Required:**
- Discrete structures
- Group theory
- Linear algebra
- Number theory
- Probability
- Statistics

### Shared Concepts and Crosscutting Themes

**Shared concepts:**
- Algorithms and Complexity
  - Cryptographic algorithms
- Architecture and Organization
  - Reverse engineering
- Artificial Intelligence
  - Machine learning models
- Data Management
  - Data security
• Graphics and Interactive Techniques
  o Privacy in XR systems
• Human-Computer Interaction
  o Usable security
• Mathematical Foundations
  o Cryptographic techniques
• Modeling
  o Access control models
• Networking and Communication
  o Secure networking protocols
• Operating Systems
  o Memory protection
• Parallel and Distributed Computing
  o Attacks due to race conditions
• Programming Languages
  o Secure compiler development
• Society, Ethics and Professionalism
  o Laws and ethics governing security and privacy
• Software Development Fundamentals
  o Defensive programming
• Software Engineering
  o Secure software engineering techniques
• Specialized Platform Development
  o Secure platform architectures
• Systems Fundamentals
  o Sandboxing techniques for isolation

Competency Specifications

• **Task 1**: Write a white paper to explain to co-workers what kinds of attacks an adversary might be able to attempt on the software being developed.
• **Competency area**: Software/Application
• **Competency unit**: Design/Development
• **Statement**: Apply adversarial thinking and systems thinking to each external interface being presented to the user.
• **Required knowledge areas and knowledge units**:
  - SEC/Foundational Security
  - DM/Data Security
  - SDF/Fundamental Programming Concepts
  - SE/Software Design
  - SE/Software Construction
  - SE/Software Verification and Validation
  - SEP/Privacy and Civil Liberties
Course Packaging Suggestions

The best way to make students aware of secure software development is to infuse security concepts into most of their coursework. However, it is likely that at least one course be offered that focuses on security so that students see the material in a holistic way. Although the breaches in computing are ongoing in the modern world, and all students should consider taking a course in security, it is unlikely that all computer science programs will be able to require such a course.

Some programs will find it easier to embed security concepts within existing courses, with Security topics blended in. For this to happen, the programs will need to understand how each of the other KAs, i.e., excluding the Security KA and the MSF KA. The guidance here to ensure every mention of security is emphasized when students are introduced to that particular concept, i.e., data security in a data management course, network security in a networking and communication courses, and so on.

Complementing this holistic could be a standalone course that offers the content listed in Foundational Security, along with some of the elective knowledge units. While the Security KA subcommittee would like to see coursework dedicated to security be in the required part of the curriculum, the members also realize that programs may not be able to do so. If so, offering such a course an elective is still important. As stated earlier, a standalone course on security may not be possible for many programs; if so, it becomes incumbent to integrate security concepts into other coursework.

Committee

Chair: Rajendra K. Raj, Rochester Institute of Technology, Rochester, NY, USA

Members:
- Vijay Anand, University of Missouri – St. Louis, MO, USA
- Diana Burley, American University, Washington, DC, USA
- Sherif Hazem, Central Bank of Egypt, Egypt
- Michele Maasberg, United States Naval Academy, Annapolis, MD, USA
- Sumita Mishra, Rochester Institute of Technology, Rochester, NY, USA
- Nicolas Sklavos, University of Patras, Patras, Greece
- Blair Taylor, Towson University, MD, USA
- Jim Whitmore, Distinguished IT Architect, USA