Ontology in Information Security: A Useful Theoretical Foundation and Methodological Tool

Victor Raskin,
Christian F. Hempelmann,
Katrina E. Triezenberg
CERIAS, Purdue University
West Lafayette, IN
vraskin, hempelma, kattriez@purdue.edu

Sergei Nirenburg
Computing Research Laboratory, New Mexico State University
Las Cruces, NM
sergei@crl.nmsu.edu

ABSTRACT
The paper introduces and advocates an ontological semantic approach to information security. Both the approach and its resources, the ontology and lexicons, are borrowed from the field of natural language processing and adjusted to the needs of the new domain. The approach pursues the ultimate dual goals of inclusion of natural language data sources as an integral part of the overall data sources in information security applications, and formal specification of the information security community know-how for the support of routine and time-efficient measures to prevent and counteract computer attacks. As the first order of the day, the approach is seen by the information security community as a powerful means to organize and unify the terminology and nomenclature of the field.

Keywords
Documentation, Security, Human Factors, Standardization, Languages, Theory.

1. ONTOLOGICAL NEEDS IN INFORMATION SECURITY. TAKE ONE
One of the many interesting results emanating from the NSPW-2000 discussions in Ballycotton was the realization that the field would gain considerably by adopting ontology as a theoretical foundation and a methodological tool. Besides my own paper on the interface between natural language processing and information security, only one other paper (Templeton and Levitt 2001—here and elsewhere, admittedly confusingly, 2001 is the year of publication of the NSPW-2000 proceedings) mentioned the term by name, but several others outlined the issues and voiced concerns, for which the ontological approach will be a valuable resource in systematizing the phenomena in the purview, enabling the modular approach, and predicting new phenomena—such as types of attack or any number of others. One give-away sign that ontology is called for is the introduction of a taxonomy and the dependence of the approach on it. Similarly, an important "side show" on anonymity at the recent IHW-01 (Pfitzmann and Köhntopp 2001) was attempting suitable and acceptable definitions for anonymity, unlinkability, unobservability, and pseudonymy and experiencing difficulties that prevented the high-powered group of researchers to reach consensus largely because of the unavailability of the ontological tool to the group. In an important initiative they call "the common language for computer security incident information," Howard and Meunier (2002) convincingly discuss the necessity to structure the incident reports to enhance rapid responses. "The two parts of this common language are

1. a set of "high-level" incident-related terms, and
1. a method of classifying incident information (a taxonomy)...

[T]he two parts of the common language (the terms and the taxonomy) are closely related. The taxonomy provides a structure that shows how most of common language terms are related. The common language is intended to help you improve your ability to

- talk more understandably with others about incidents,
- gather, organize, and record incident information,
- extract data from incident information,
- summarize, share, and compare incident information,
- use incident information to evaluate and decide on proper courses of action, and
- use incident information to determine effects of actions over time."

This passage summarizes very well what an ontology for the domain of information security can do because, coupled with the ontology-based lexicon, it provides "the two parts of the common language" for the field, and much more.

1. WHAT IS ONTOLOGY?
Not to be confused with the philosophical discipline of metaphysics, long the laughing stock of empiricist philosophy and recently experiencing a spectacular comeback, ontology is a constructed model of reality, a theory of the world—more practically, a theory of a domain. In still more practical terms, it is a highly structured system of concepts

NSPW'01, September 10-13th, 2002, Cloudcroft, New Mexico, USA.

covering the processes, objects, and attributes of a domain in all of their pertinent complex relations, to the grain size determined by such considerations as the need of an application or computational complexity. Thus, an ontology may divide the root concept ALL into EVENTs, OBJECTs, and PROPERTYs (Fig. 1); EVENTs into MENTAL-EVENTs, PHYSICAL-EVENTs, and SOCIAL-EVENTs (Fig. 2); OBJECTs into INTANGIBLE-OBJECTs, MENTAL-OBJECTs, PHYSICAL-OBJECTs and SOCIAL-OBJECTs (Fig. 3); PROPERTYs into RELATIONs (bi- or multiplace attributes) and ATTRIBUTEs (one-place) (Fig. 4, 5)—and so on, to finer and finer details.

Figure 1. ALL tree, 1 level down.

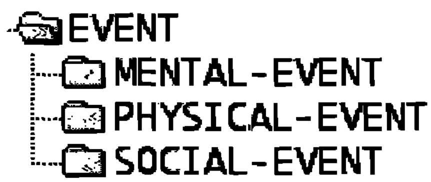

Figure 2. EVENT tree, 1 level down.

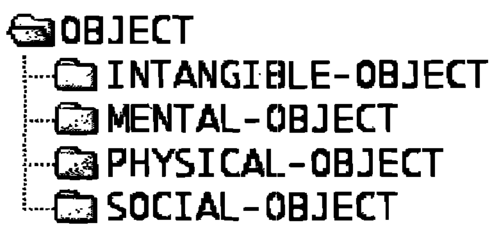

Figure 3. OBJECT tree, 1 level down.

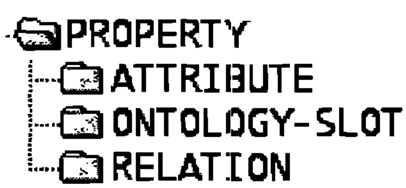

Figure 4. PROPERTY tree, 1 level down.

Formally, then, an ontology is a tangled hierarchy of conceptual nodes, each of which can be represented as:

concept-name
    (property-slot property-value)+

In other words, a concept has one or (usually) more properties. Every concept but the root ALL has the property IS-A, and the value of the property is the parent of this concept, the higher node—so the concept MENTAL-PROCESS, a child of PROCESS, is, on partial view, as follows:

mental-process
    is-a   process
        (property-slot property-value)+

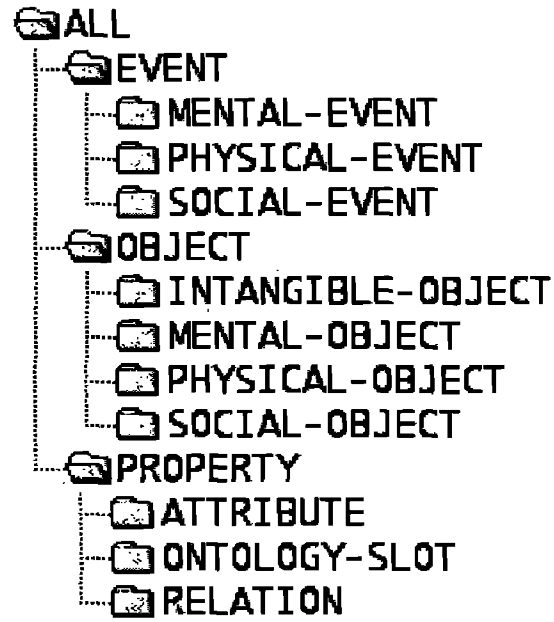

Figure 5. ALL tree, 2 levels down.

The value of the IS-A property may be a disjunction of two or more concepts. Thus, a concept may have multiple parents and multiple inheritance. It shares the latter formal feature with the object-oriented programming languages, which are indeed suitable for implementing ontological procedures. The object-oriented approach lacks the conceptual content of ontology, so it is not sufficient for addressing the information security needs discussed here. To our (limited) knowledge, no object-oriented proposal of this kind has been made. The distinction between form and content is crucial for understanding the proposed ontological paradigm, and it often escapes the formalism-based disciplines. The discussion at the Workshop contributed significantly to clarifying this distinction, and we hope that this article is the next step in the same direction. It is also possible to present this format of ontology as a lattice—in fact, the ontologies constitute a special subset of lattices. Again, however, it is the content of ontologies that makes them useful for information security, independently of the choice of formats.

Obviously, an ontology provides a powerful taxonomic tool for an unlimited set of phenomena because each property-slot determines the class of concepts that have the property and each value a subclass of that class. A typical ontology has hundreds of properties. It is noteworthy also that, with an ontology (as with the object-oriented approach), one escapes the problems of cross-classification, when deciding which of, say, the two features to apply first has a theoretical and methodological price tag.

But an ontology is much more than that—primarily because of inheritance. Inheritance is the down-propagation of properties, with their values filled, from parents to children and further descendants. When we look at a table we may notice that it is made of wood, is oval-shaped, and has four legs. Each of these property values could be different with a different table, so these properties belong to this particular object. But we know much more about the properties of this table: We know that it is designed to be used for various purposes, usually in a room, usually long-term, usually rather expensively, and it would have been bought in a furniture store—all of that we know by virtue of a table being furniture, i.e., the concept FURNITURE is a parent of the concept TABLE. We also know that the table was specially manufactured by a human or humans (who may have designed and/or operated machines in the process of manufacturing the table) rather than being a naturally occurring object—this we know because TABLE inherited that property from ARTIFACT, the parent of FURNITURE. Finally, we know that the table has three spatial and one temporal dimension, i.e., that the table occupies a certain space at a certain time—because its ontological ancestor ARTIFACT is a child of PHYSICAL-OBJECT.

This simple example of how the various properties originate with the concept itself or are inherited from an ontological ancestor can be repeated with computer attacks or any other types of phenomena, not necessarily related to natural language and certainly independent of any specific language, and every participant can produce such examples from his or her own research purview. In fact, we would challenge any participant to declare and defend a view that his or her approach has no ontological material in it. We, on the other hand, would like to be challenged to demonstrate the benefits of the ontological resource for any approach, and we would proceed to do so by asking the challenger a short list of pertinent questions about the nature of the phenomena the approach deals with. Any similarity to the composition problem, a scary prospect for an otherwise most well-disposed anonymous reviewer, is not intended here and, we believe, not present, and the discussion did not bring up any unfamiliar formulation of that problem.

3. ONTOLOGICAL NEEDS IN INFORMATION SECURITY. TAKE TWO

What we are proposing here is extending research and application paradigms in information security by including natural language data sources. The proposal concentrates on two issues:

- Inclusion of natural language data sources as an integral part of the overall data sources in information security applications, and
- formal specification of the information security community know-how for the support of routine and time-efficient measures to prevent and counteract computer attacks

Where does natural language data play a role in InfoSec? Here are some representative examples:

- sysadmin logs are written in a sublanguage of a natural language (and can be allowed to contain more complex language if the processing systems are capable of treating it);
- information hiding (steganography, NL watermarking) depends on NLP;
- downgrading will provide automatic filtering of sensitive information from documents intended for dissemination;
- documents in natural language can be scanned for detecting possible intellectual property leakage;
- if an InfoSec task involves human alongside software agents, NLP is the most efficient way of interagent communication.

In the past, the above tasks, if at all attempted, were supported by either keyword-based search technology or through stochastic mechanisms of matching and determination of differences between two documents. These approaches have approached the ceiling of their capabilities.

We propose a new, content-oriented, knowledge- and meaning-based approach to form the basis of the NLP component of the information security research paradigm. The difference between this knowledge-based approach and the old "expert system" approach is that the former concentrates on feasibility, for example, by using a gradual automation approach to various application tasks. The ontological approach also deals, however, albeit at a much more sophisticated level with encoding and using the community know-how for automatic training and decision support systems. The cumulative knowledge of the information security community about the classification of threats, their prevention and about defense against computer attacks should be formalized, and this knowledge must be brought to bear in developing an industry-wide, constantly upgradeable manual for computer security personnel that may involve a number of delivery vehicles, including an online question-answer environment and a knowledge-based decision support system with dynamic replanning capabilities for use by computer security personnel. The underlying knowledge for both of these avenues of information security paradigm extension can, as it happens, be formulated in a single standard format. The knowledge content will readily enjoy dual use in both NL data inclusion and decision support, and it is made possible through the use of ontologies. Fig. 6 below shows a generic scheme of interaction of the ontological resources applied to a conceptual domain, such as information security.

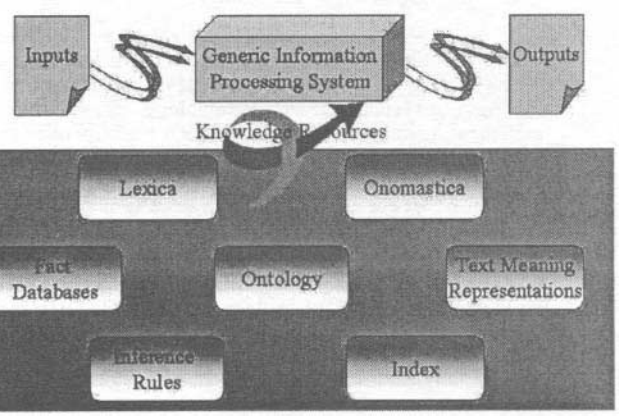

Figure 6. Application of the Ontological Paradigm to a Domain .

The ontological paradigm is already used at NMSU CRL to support such basic NLP tasks as machine translation, information retrieval and extraction, question answering, planning and summarization. These tasks have been integrated at CRL and CERIAS, as well as other sites, such as Bell Labs, in end applications for data mining, information security, intelligence analysis, etc.

We will now elaborate a bit on the three major benefits mentioned at the beginning. First, ontology organizes and systematizes all the phenomena in the research purview (such as types of computer attack) at any level of detail, and reduces a large diversity of items to a much smaller list of properties. Secondly, most approaches gain from induced modularity, for instance, by relating certain measures to the detection of certain properties (e.g., if a certain property of an attack calls for a certain measure, a complex attack, with a set of properties, will call for the corresponding set of countermeasures). Third, by providing the full combinatorics of the compatible properties, an ontologically-based approach may predict additions to its purview (for instance, possible types of attack that have not ever occurred yet).

There are additional benefits to the implementability of ontology within an approach. Ontology lends itself easily to an expansion, such as the addition of a new property, without any modification of the existing ones. Of course, the addition of a new concept is an even easier thing. (A small pilot project on extending the existing ontology to the field of information security is, in fact, already underway at CERIAS.) A highly formal object, ontology can be presented in the pseudocode, BNF, or other appropriate formalisms that lend themselves more easily to programmability and computability. The current stage of development in ontology makes a number of important ready-made resources available to the researcher or practitioner. These include:

- ready-made ontologies, general or for specific domains;
- formalisms, techniques, and interfaces for importing ontologies;
- automatic and semi-automatic tools for detecting and acquiring new properties;
- instrumentation for acquiring new concepts within a domain;
- techniques for identifying and adding a new domain or subdomain to an ontology.

It is noteworthy that, while intrigued by all those possibilities, the Workshop participants felt that the first order of the day was to use the ontological approach to firm up and unify the concepts and terminology. We are already implementing this task within a CERIAS/Eli Lilly pilot grant at Purdue University, starting from a glossary of terms in Appendix 2. Appendix 1 contains some discussion and examples of lexical and ontological entries acquired with that project.

2. CONCLUSION

We have achieved considerable progress on the interface of natural language processing and information security (see Raskin et al. 2001; Atallah and Raskin 2001) on the basis of these ontological resources, and natural language involves much more complex ontologies than many areas of information security require. This makes us think that the community should discuss ontology as an extremely promising new paradigm in the field. I hope that an energetic discussion of the topic will support, enrich, and specify this view and lead to collaborative research on the use of ontology.

3. ACKNOWLEDGMENTS

This paper started out as a discussion proposal. The discussion at the Workshop generated a lively discussion, and the paper reflects the issues discussed and the answers to the questions. It also reflects the more advanced stage of research on the ontology for information security. The authors are grateful to the editors of this volume for their understanding that an updated version of the paper will serve the community better. We are also grateful to the Eli Lilly Foundation and to CERIAS for making the finds for the pilot grant available. We greatly appreciate the discussants' contributions at the Workshop and Bob Blakley's incredibly detailed and accurate rendition of it for our benefit. We owe a special debt of gratitude to the Workshop leadership, in particular, Steve Greenwald and Cristina Serban, for the unprecedented special permission to Sergei Nirenburg, from his nearby base in Las Cruces, NM, to join Victor Raskin for the presentation of the proposal—his energy made it much more successful.

4. REFERENCES

[1] Atallah, M., and Raskin, V. Natural language watermarking: Design, analysis, and a proof-of-concept implementation. In: Moskowitz, I.S. (ed.). Pre-Proceedings of the 4th Information Hiding Workshop. Pittsburgh University Center, Pittsburgh, PA, 2001, 193-208. See also http://chacs.nrl.navy.mil/IHW2001/accepted.html or http://omni.cc.purdue.edu/~vraskin/IHW.AtaRasEtAl.pdf).

[2] Kabay, M. and Bosworth, S. (eds.). Computer Security Handbook, 4th ed. John Wiley and Sons, New York, NY, 2002.

[3] Pfitzmann, A., and Köhntopp, M. Anonymity, unobservability, and pseudonymy—A proposal for terminology, Position paper for a symposium on anonymity at IHW-01, 2001. http://www.koehntopp.de/marit/pub/anon/ihw/Anon_Terminology_IHW.pdf.

[4] Raskin, V., Atallah, M., McDonough, C., and Nirenburg, S. Natural language processing for information assurance and security: An overview and implementations. In: Proceedings of NSPW-2000. ACM Press, New York, NY, 2001, 51-65.

[5] Templeton, S., and Levitt, K. A requires/provides model for computer attacks. Ibid, 31-38.

5. ADDITIONAL RESOURCES

For a detailed description of the largest fully implemented ontology, see Chapter 7 of S. Nirenburg and V. Raskin's Ontological Semantics, forthcoming, http://crl.nmsu.edu/Staff.pages/Technical/sergei/book/index-book.html. To browse the Web tool for largely the same ontology, go to http://messene.nmsu.edu:9021/, guest login "purdue," guest password "ont590" (sorry, no editing privileges).

For other useful sites on ontology, check out "Links to Other Ontology Sites" at http://crl.nmsu.edu/Research/Projects/mikro/htmls/ontology-htmls/onto.index.html.

See also www.fois.org for an important forthcoming conference, where some of the similar positions will be presented to the ontologists.

8. APPENDIX

8.1 Examples of Entries

As shown on Fig. 6 above, ontology and lexicons are two of the static resources within the ontological semantic paradigm. Fig. 7 shows an entry for computer-security, clearly displaying both its own, locally defined properties and the ones inherited from its ancestors.

⊞ Defined in **COMPUTER-SECURITY**		
⊞ DEFINITION	VALUE	a field that develops software to assure and secure information and protect against unauthorized access
⊞ IS-A	VALUE	⊞ COMPUTER-SCIENCE, ⊞ SOFTWARE-ENGINEERING
⊞ Inherited from **FIELD-OF-STUDY**		
⊞ THEME-OF	SEM	⊞ ACTIVE-COGNITIVE-EVENT
⊞ HAS-PARTS	SEM	*NOTHING*
⊞ PART-OF	SEM	*NOTHING*
⊞ Inherited from **ABSTRACT-OBJECT**		
⊞ CAUSED-BY	SEM	*NOTHING*
⊞ Inherited from **MENTAL-OBJECT**		
⊞ PATH-OF	SEM	⊞ CHANGE-LOCATION, ⊞ EVENT

Figure 7. Ontological entry for computer-security .

Ontology is, of course, language-independent, i.e., it is the same for all languages. An ontological lexicon is, on the contrary, language-dependent, i.e., each language requires its own lexicon, containing its own words and phrasals—the same meanings, however, will be present in the lexicons but distributed differently among words. The English lexicon contains a lexical entry for one sense of the word anonymous (Fig. 8); this same meaning will appear in the lexicons for other languages, where it will be one of the senses of other words, such as *anonyme* in French, *anonimnyy* in Russian, *anonimi* in Hebrew, etc.

In the entry, the syn-struc part defines the two syntactic patterns, in which the adjective—and virtually all English adjectives—may occur, namely, the attributive, as in [it is an] anonymous message, and predicative, such as [this message] is anonymous.

```
Anonymous-Adj1
    cat        adj
    syn-struc    1    root     $var1
                      cat      n
                      mods     root     anonymous
                 2    root     big
                      cat      adj
                      subj     root     $var1
                               cat      n
    sem-struc
          12
                      ^$var1
                               sem      event
                               agent *unknown*
```

Figure 8. English lexical entry for *anonymous*.

8.2 Glossary Items Being Added to the Ontology and English Lexicon

To adjust the latest implementation of the ontology to the domain of information security, we have been implementing the first stage of checking and adapting the existing concepts as well as acquiring new concepts in the ontology part of the pilot grant project and checking and adjusting lexical entry senses as well as acquiring new entries in the English lexicon. Below is the list of the words and phrasals to be acquired by the conclusion of the project in August 2002. For each item on the list, we make sure that there is an entry in the English lexicon with the appropriate sense and that the concepts required for defining such an entry are in place in the ontology.

The list has been compiled from the indices of standard introductions to the field of information security as well as some existing glossaries that were available to us. The list does not claim to be fully representative, let alone exhaustive, and it is printed here to:

- give the community a sense of the scope of the current project, and
- to solicit suggestions for additional sources as well as individual items for inclusion.

Absolute rate
Access control
Access control list
Access control matrix
Access log
Access triple
Accountability
accuracy
Address
Adjudicable
Aggregate query
Aggressive scheduler
Algorithm
Amateur
Analog
Analyzability
Anklebiter
Anonymity
applet
Arbiter
AS-400
Associativity
Assurance
Assymetric encryption
Attack
Attribute
Audit
Audit log
Audit options
Authenticate
Authentication
Authenticity
Automatic retaliation
Availability
Backdoor
Backup
Base register
Bastion host
Block cipher
Boot sector virus
Bootstrap virus
Bounds register
Break
Brute force attack
Buffer
Buffer overflow
Caesar cipher
Call bracket
Capability
Career criminal
category
CERT
Certificate
Certificate distribution center
Certificate revocation list

Certification authority
Certified code
Certified mail
CGI script
Change log
Channel
Checksum
Chinese wall policy
ChineseWall Model
cipher
Cipher block chain
Ciphertext
Classification
Clearance
Client
Clique problem
Code
collision
Columnar transposition
Commit
Commitment
Common criteria
Commutativity
Compartment
Complexity
Composite
Compression
Computing system
Conceal
Concurrency-control
Confidentiality
Configuration management
Confusion
Connectivity
Conservative scheduler
Constrained data item
Contract signing
Control
Controlled sharing
Cookie
Copy
Copyright
CORBA
Core
Core dump
Correct
Coupling
Cover story
Covert
Covert channel
Covert timing channel
Cracker
credentials
Criteria creep
cryptanalysis
Cryptanalyst
Cryptography
Cryptology
Cryptosystem
Cycle
Data
Data encryption standard
Database
Database management system
Datagram
Decidability
Decipher
Decode
Decrypt
Degausser
dependability
Diagram
Diffusion
Digest
Digital
Digital signature
Digital signature scheme
Directory
Disaster
Disclosure
Distributivity
Divisible by
Domain
Dominance
Dongle
Double transposition
Driver
Effectively secure
Effectiveness
Egoism
Egoless programming
Electronic-code-book-mode
Element
Encapsulation
Encipher
Encode
Encrypt
Equivalent
Error code
Error propagation
Ethic
Etiquette
Evaluation
Evidence
Executive
Exhaustive attack
Expandability
Exposure
Fabrication
Fair use
Fairness
Fence reigster
Field
Field check
File protection
Filter
Fire
Firewall
Flood
Flooding
Frequency distribution
Front end
Guard
Guest
Hack
Hardware
Hash
Heat
Hierarchy
Host
Identity
Impersonate
Index of coincidence
Inductance
Inference
Information
Information hiding
Information leak
Integrity
Integrity
Intercept
Internal consistency
Interpretation drift
Interruption
Intruder
Inverse divide
Inverse mod
Isolation
Join
Kasiski method
Kernel
Key
Key distribution server
Keyless cipher
Knapsack
Lattice model
Layering
Least privilege
License
Limited privilege
Link
Local name space
Logic
Logic analyzer
Logic bomb
Lucifer
Macro
Macro virus
Maintain
Malicious code
Master key
Measure of roughness
Mechanism
Memory-resident virus
Mental poker
Message digest
Microwave
Modem
Modification
Modular arithmetic
Module
Modulus
Monitor
Monoalphabetic cipher
Multiplex
Mutual suspicion
Need-to-know
Network
Node
Nondeterminism
Notarization
Notary
Novelty
Nucleus
Object
Object request broker
Oblivious transfer
One-time
Open design
Optical fiber
Oracle machine
Originality
Packet
Packet sniffer
Paging
Parasitic virus
Parity
Password
Patent
Payload
Peer code review
Peer design review
Permission
Permutation
PGP (pretty good privacy)
Physical
Plaintext
Policy
Polyalphabetic cipher
Polymorphic (virus)
Polynomial
Port
Precise
Prime number
Privacy
Probable password
Problem
Product cipher
Program
Project
Property
Protect
Protected object
Protocol
Query
Rabbit
Random access memory
Read only memory
Receiver
Record
Recover
Reducibility
Redundancy
Relation
Relative prime
Reliable
Religion
Relocation
Repeater
Replay
Resident virus
Resident virus

Resource
Reuse
Reverse engineer
Ring bracket
Risk
Rogue program
Routing
Salami attack
Satellite
Satisfiability problem
Schema
Secrecy
Secure
Security audit
Segment
Segmentation
Self-enforcing protocol
Semantic sugar
Sender
Sensitive
Sensitive data
Separation
Server
Service program

Session
Session key
Shadow program copy
Shared file
Shared resource matrix
Shell theft
Shredder
Shrink-wrapped software
Side effect
Simple substitution
Single-user system
Socket
Software
Solvable problem
spoof
Stream cipher
Stub
Subject
Subscheme
Substitutions
Suppress
Surge
Symmetric
Symmetric key exchange

tamper
Tamperproofness
Target
Temporal
Terminal
Test
Theft
Threat
Time bomb
Time stamp
Topology
Trade secret
Traffic key
Transformation procedure
Transient virus
Transmission medium
Transposition
Trapdoor
Trigram
Tripwire
Trojan horse
Trusted
Unbypassability
Unconditionally secure

Understand
Unicity distance
Unix
Usage restriction
User
Validation
Verification
Vernam cipher
View
Vigenere tableau
Virtual
Virtualization
Virus
Virus scanner
Virus signature
Vulnerability
Window
Wiretap
Workstation
Worm
Write-down