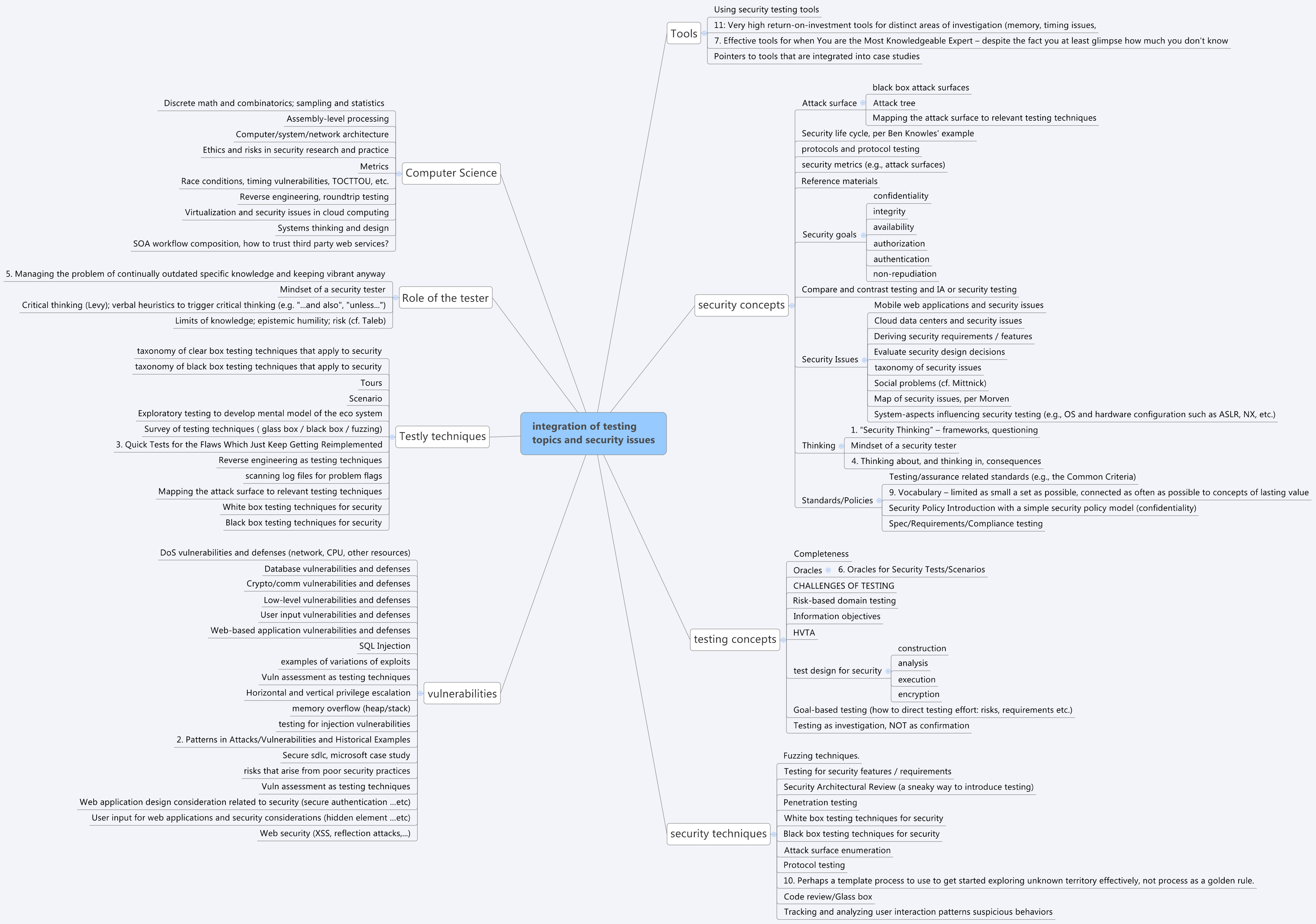

integration of testing topics and security issues

- Tools
  - Using security testing tools
  - 11: Very high return-on-investment tools for distinct areas of investigation (memory, timing issues,
  - 7. Effective tools for when You are the Most Knowledgeable Expert – despite the fact you at least glimpse how much you don't know
  - Pointers to tools that are integrated into case studies
- security concepts
  - Attack surface
    - black box attack surfaces
    - Attack tree
    - Mapping the attack surface to relevant testing techniques
  - Security life cycle, per Ben Knowles' example
  - protocols and protocol testing
  - security metrics (e.g., attack surfaces)
  - Reference materials
  - Security goals
    - confidentiality
    - integrity
    - availability
    - authorization
    - authentication
    - non-repudiation
  - Compare and contrast testing and IA or security testing
  - Security Issues
    - Mobile web applications and security issues
    - Cloud data centers and security issues
    - Deriving security requirements / features
    - Evaluate security design decisions
    - taxonomy of security issues
    - Social problems (cf. Mittnick)
    - Map of security issues, per Morven
    - System-aspects influencing security testing (e.g., OS and hardware configuration such as ASLR, NX, etc.)
  - Thinking
    - 1. "Security Thinking" – frameworks, questioning
    - Mindset of a security tester
    - 4. Thinking about, and thinking in, consequences
  - Standards/Policies
    - Testing/assurance related standards (e.g., the Common Criteria)
    - 9. Vocabulary – limited as small a set as possible, connected as often as possible to concepts of lasting value
    - Security Policy Introduction with a simple security policy model (confidentiality)
    - Spec/Requirements/Compliance testing
- testing concepts
  - Completeness
  - Oracles
    - 6. Oracles for Security Tests/Scenarios
  - CHALLENGES OF TESTING
  - Risk-based domain testing
  - Information objectives
  - HVTA
  - test design for security
    - construction
    - analysis
    - execution
    - encryption
  - Goal-based testing (how to direct testing effort: risks, requirements etc.)
  - Testing as investigation, NOT as confirmation
- security techniques
  - Fuzzing techniques.
  - Testing for security features / requirements
  - Security Architectural Review (a sneaky way to introduce testing)
  - Penetration testing
  - White box testing techniques for security
  - Black box testing techniques for security
  - Attack surface enumeration
  - Protocol testing
  - 10. Perhaps a template process to use to get started exploring unknown territory effectively, not process as a golden rule.
  - Code review/Glass box
  - Tracking and analyzing user interaction patterns suspicious behaviors
- Computer Science
  - Discrete math and combinatorics; sampling and statistics
  - Assembly-level processing
  - Computer/system/network architecture
  - Ethics and risks in security research and practice
  - Metrics
  - Race conditions, timing vulnerabilities, TOCTTOU, etc.
  - Reverse engineering, roundtrip testing
  - Virtualization and security issues in cloud computing
  - Systems thinking and design
  - SOA workflow composition, how to trust third party web services?
- Role of the tester
  - 5. Managing the problem of continually outdated specific knowledge and keeping vibrant anyway
  - Mindset of a security tester
  - Critical thinking (Levy); verbal heuristics to trigger critical thinking (e.g. "...and also", "unless...")
  - Limits of knowledge; epistemic humility; risk (cf. Taleb)
- Testly techniques
  - taxonomy of clear box testing techniques that apply to security
  - taxonomy of black box testing techniques that apply to security
  - Tours
  - Scenario
  - Exploratory testing to develop mental model of the eco system
  - Survey of testing techniques (glass box / black box / fuzzing)
  - 3. Quick Tests for the Flaws Which Just Keep Getting Reimplemented
  - Reverse engineering as testing techniques
  - scanning log files for problem flags
  - Mapping the attack surface to relevant testing techniques
  - White box testing techniques for security
  - Black box testing techniques for security
- vulnerabilities
  - DoS vulnerabilities and defenses (network, CPU, other resources)
  - Database vulnerabilities and defenses
  - Crypto/comm vulnerabilities and defenses
  - Low-level vulnerabilities and defenses
  - User input vulnerabilities and defenses
  - Web-based application vulnerabilities and defenses
  - SQL Injection
  - examples of variations of exploits
  - Vuln assessment as testing techniques
  - Horizontal and vertical privilege escalation
  - memory overflow (heap/stack)
  - testing for injection vulnerabilities
  - 2. Patterns in Attacks/Vulnerabilities and Historical Examples
  - Secure sdlc, microsoft case study
  - risks that arise from poor security practices
  - Vuln assessment as testing techniques
  - Web application design consideration related to security (secure authentication ...etc)
  - User input for web applications and security considerations (hidden element ...etc)
  - Web security (XSS, reflection attacks,...)