Basic Principles of Security Engineering. Tom@Gilb.com

For VA

1. Security is only one of many critical stakeholder requirements of your system. Security needs to balance its needs with other stakeholder priorities.
2. You can only understand how much Security you can realistically plan to deliver to a system, by knowing rather specifically, about the levels of *all other* stakeholder priorities: Balance.
3. A Systems Architect is one name for the instance that co-ordinates and balances all competing stakeholder needs, including security.
4. An engineering approach is necessary, to model large and complex systems, and to find a good balance for Security. This includes quantifying all quality requirements and other variable stakeholder values, and limited resources: in the short term and for the system lifetime.
5. Systems and Security Engineering must include a means of both estimating, and measuring the multiple impacts on all critical stakeholder values (qualities and other values) and life-cycle system-resource consumption. (Hint see Gilb Impact Estimation Table, book Competitive Engineering 2005)
6. The safest proven approach (See IBM Cleanroom, Quinnan) for complex systems engineering will attempt to deliver small (2% of budget) incremental steps of the security design, measure actual security levels attained, change design when increments fail to deliver enough, and stop when target levels are delivered. This is Agile Security Engineering.
7. A security design can be absolutely anything, not violating stated system constraints, which gives the best security impact in the direction of our numeric security targets, with the least consumption of budgeted resources. ‘Security Efficiency’. Anything means, any design, from any effective discipline, for the system.
8. Security requirements can state minimum levels (constraints, worst case), and more-valuable, more-desired levels (target levels). We should be able to explain the difference (Target Level - Constraint Level) in terms of consequential loss dimensions, such as costs, if we do not attain the target levels. These 2 levels help the security engineer and the systems architect determine current priorities as system development progresses. For example when targets are reached the security or other quality dimension loses all current priority.
9. Security engineers need to co-operatively recognize that security itself is ultimately dependent on many other qualities of the system also being attained, at high interesting levels, for example usability, safety, reliability, availability, work capacity, trustworthiness, adaptability, portability, maintainability, recoverability and many others.
10. Security engineering and maintenance of good security levels is a never-ending lifetime battle, not a one -time up-front design effort; so persistent resources to monitor the security threats, and necessary security levels, must be a part of the lifetime operational costs, of any large and complex system.

Maybe ten principles is enough for the moment, but my books Principles of Software Engineering Management (1988), Competitive Engineering (2005) and Value Planning (2014-2019 Digital), see [gilb.com](http://gilb.com) provide hundreds more, deep and long-lasting principles of systems engineering.