1 Title

2 Networks in the Bad Old Days

- Large networks
- many devices needed individual management
- configuration version control unheard of (well, mostly)
- expensive to manage and easy to mess up

3 The Brave New World!

- Centralized management
- more responsive potentially
- easy (-ier) to manage
- potentially more responsive
- applications, controllers, repositories combine to provide better infrastructure control

4 Brave New implications

(see slides)

5 host $\Longleftrightarrow$ switch

- Confidentiality not critical
  - switches don't consume host services
  - hosts don't confirm delivery by a specific switch (usually)
  - no *explicit* control traffic
- Integrity vital
  - hosts expect reliable data plane information transfer
  - switches don't communicate to hosts
- Availability expected
  - Hosts do want networks up when they need them, but a data plane concern
- Non-repudiation
  - Hosts don't expect non-repudiation of traffic handling
  - Switches don't handle non-repudiation of handled traffic (though other devices might to this, forensics, auditing, etc.)
- Authentication
  - Hosts generally don't need to authenticate a switch, though with very sensitive traffic this may be important
  - Switches may authenticate hosts when doing data plane traffic prioritization or service shaping

6 switch $\Longleftrightarrow$ controller

- Confidentiality only important in some edge cases
  - Not generally important as the effects of any controller messages is visible
  - Defense-oriented messages (e.g. related to malicious traffic redirection) may need to be confidential; this implies all traffic kept confidential in these cases
- Integrity again vital
  - Switches need to be able to trust controller messages
  - Controllers will very likely issue commands to specific switches based on switch status, so messages from switches to controllers must be trustworthy
- Availability paramount
  - Controllers must be available to switches for SDN to work correctly; otherwise, switch behavior is undefined, though they will usually use the most recent flow table for a while
  - Switch availability is much less important from a control-plane perspective
- Non-repudiation perhaps more important
  - Useful for forensics, network debugging, to see which controllers issued which commands and past switch status
  - Useful for trust measures
- Authentication of controllers important
  - Switches should be able to authenticate a given controller to establish controller authority
  - Controllers may only need to authenciate switches in high security environments, to assure that all managed switches are in fact authorized to handle traffic on a network

7 controller $\Longleftrightarrow$ repository

Important to note, this model can be implemented via application access to controllers as well

- Confidentiality not always vital
  - Controllers may need to access repositories for global state information
  - Repositories may send information to controllers
  - In either case, the information can generally be derived from other sources
  - Security-related information may be important in some cases
- Integrity important, as usual
  - Controllers need to be able to trust actionable information from repositories
  - Repositories generally have no data dependencies on controllers, though they may in cases where controllers report status to enable global network awareness
- Availability not vital
  - Controllers can make local decisions that can aggregate into a nearly optimal global state
  - Global state information becomes more important at scale as the cost of cumulative inefficiency grows
- Non-repudiation a big less important for core control plane functions
  - Again most useful for trust evaluation and forensics
- Authentication of repositories important
  - If a controller is going to use repository supplied information to make control decisions, the source of that information must be trustworthy
  - Likewise, repositories collecting information from controllers (or other sources) must be able to have confidence that the information delivered is from a source that can be trusted

8 controller $\Longleftrightarrow$ application

Applications are generalities of the previous repository construct, but run into issues when we have multiple applications accessing a given controller

- Confidentiality based on application
  - Confidentiality requirements are really based on the application type and the information submitted to the controller; e.g. logging apps may not be as important as security apps
- Integrity important, as usual
  - Controllers need to be able to trust actionable information, and applications need to be able to trust controllers for data collection
- Availability not vital
  - Heavily dependent on the specific application
- Non-repudiation needs again based on application
  - Most useful for trust evaluation and forensics
- Authentication of certain applications important
  - If an app can influence a controller, it must be authenticated
  - If a controller submits data to an app, it must be authenticated

9 Attribute Commonality

- Confidentiality
  - Usually information can be derived from network behavior
  - Exceptions for cyber-security use cases; these can drive confidentiality needs into entire system
- Integrity
  - Actionable information leading to control-plane decisions needs to be trustworthy
- Availability
  - Of varying importance
  - Most important in switch $\Longleftrightarrow$ controller relationships
- Non-repudiation
  - Not that important, usually relegated to trust or forensics use cases
  - Cheap to add if we already have authentication and message integrity though
- Authentication
  - Related to integrity
  - Control agents (controllers, etc.) need to be able to establish that a source is trustworthy as well as that the information from that source has not been tampered with

10 Differentiating Attributes of SDN

Compared to other more agent-centric systems, SDN control systems have some advantages:

- Limited control-plane volatility
  - MANETs and agent-based systems are much more chaotic with respect to functional distribution (many devices wear multiple communication hats) and suffer from frequent attach / detach issues
- Centralized High-Availability
  - Any high-availability requirements are constrained to specific functional areas (e.g. controllers)
- Clearly Defined Roles
  - SDN entities have clear roles; systems in MANETs or agent-based systems frequently do not
- Predicable Expected Behavior
  - Clearly defined roles should lead to more predicable behavior and correspondingly easier behavioral outlier detection

11 Comments and Questions

(see slides)