INFRASTRUCTURE SECURITY

this security is provided by the administrative authority (AA) of a network, on behalf of itself, its customers, and its legal authorities

Goals

* **prevent or mitigate resource attacks (one way to make a denial-of-service attack)**

  - **resources are link bandwidth, compute cycles, memory**

  - **there is always some amplification (or the attacker would be no better off than the target)**
    e.g., botnets
    e.g., SYN flood
    e.g., random subdomain DNS attack
    e.g., demanding HTTP requests
    e.g., reflection attack
    e.g., trigger Ethernet broadcast

  - **the attacker often tries to hide**
    e.g., false IP source address ("spoofing")
    e.g., master of a botnet is hidden
    e.g., short times-to-live

  - **because of fate sharing, an attack at any level works**

Goals, Continued

* block specific communications
  - spam, robocalls, malware
  - illegal communication, communication that violates parental controls (might be recorded instead of blocked)
  - unplanned communication in an enterprise network
  - port scanning
* protecting freedom and privacy
  - complementary to endpoint security, because the adversary observes packet headers, etc. that are not encrypted
  - the opposite of blocking—-same technology, taking different sides

Packet Filtering

* by far the most common technique for infrastructure security

BASIC FILTERING

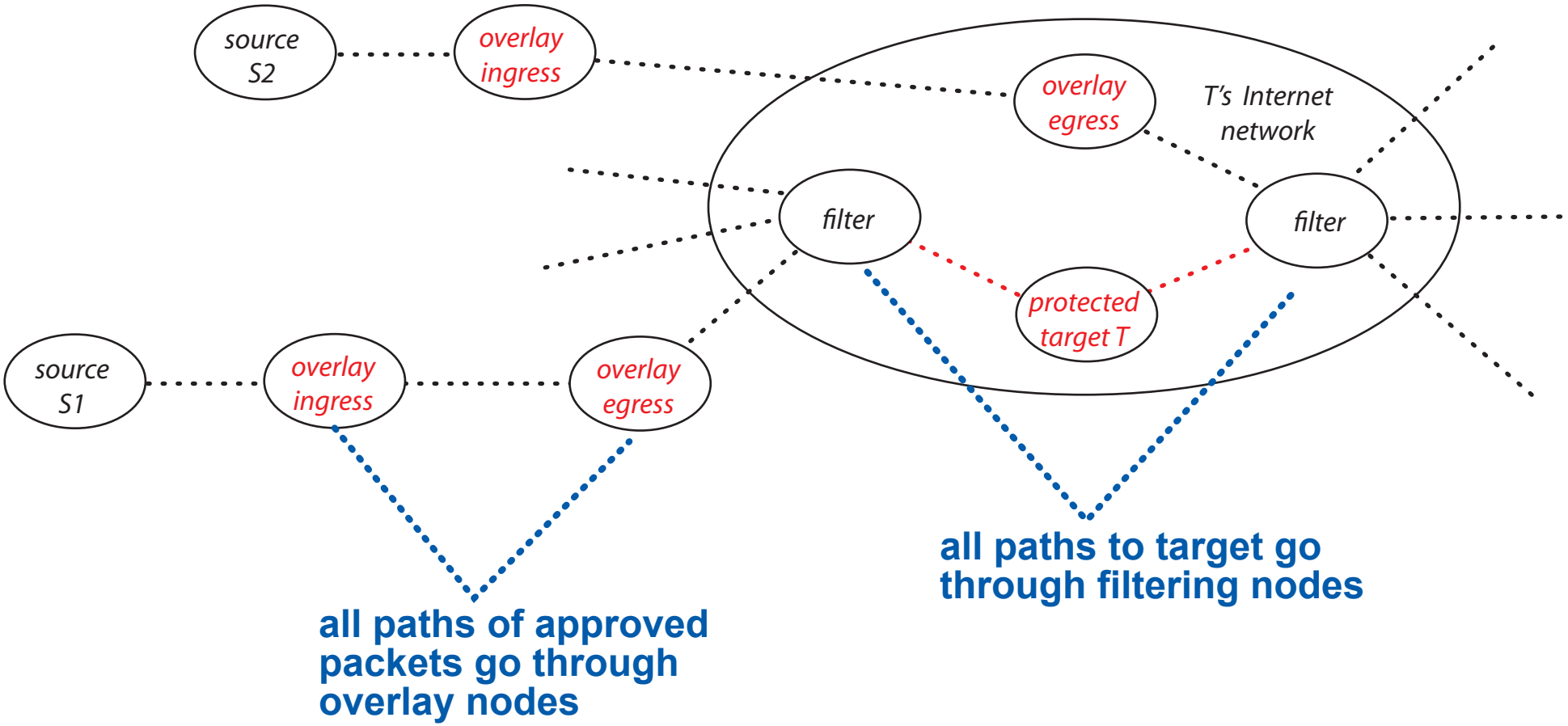
	FIREWALL	ROUTER	INTRUSION DETECTION SYSTEM	INTRUSION PREVENTION SYSTEM
FILTERING CRITERIA	predicates on IP packet headers can have a table of ongoing sessions		any predicates; keeping data on traffic, looking for anomalies	
ACTIONS TAKEN	drop packets		raise an alarm, divert packets for further analysis	drop packets, refuse requests
PACKET STEERING	located at all network edges, need session affinity	located on all packet paths	special-purpose forwarding	
HOW ARE THE FILTERS PROTECTED?	big capacity		virtualized for dynamic scale-out	

* filtering criteria
  - signature-based
  - source name
    if bad communication (flooding, spam) is one-way, source name can be false, which is a problem; on the other hand, having a false source name is a clear sign of bad communication
  - anomaly detection for diagnosis of flooding attacks
  - filtering criteria are a real weak point in infrastructure security; either attacks are missed or there is a lot of collateral damage from filtering out good packets

* positive filtering: drop is default, identify good packets to get through
  - there is an overlay pattern for positive filtering

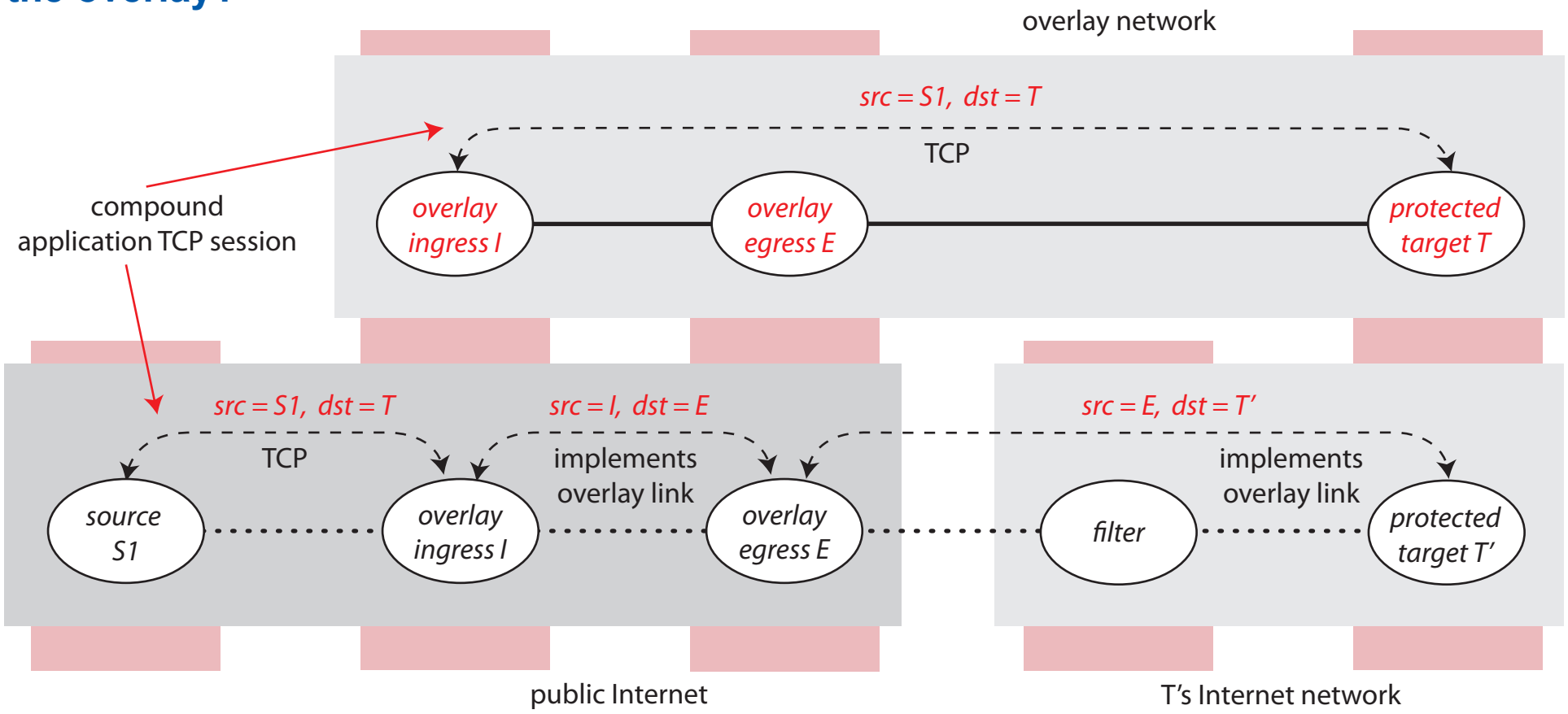
OVERLAY PATTERN FOR POSITIVE FILTERING

PATHS THOUGH MACHINES TO PROTECTED TARGET



OVERLAY PATTERN FOR POSITIVE FILTERING

1. How do packets get approved and enter the overlay?



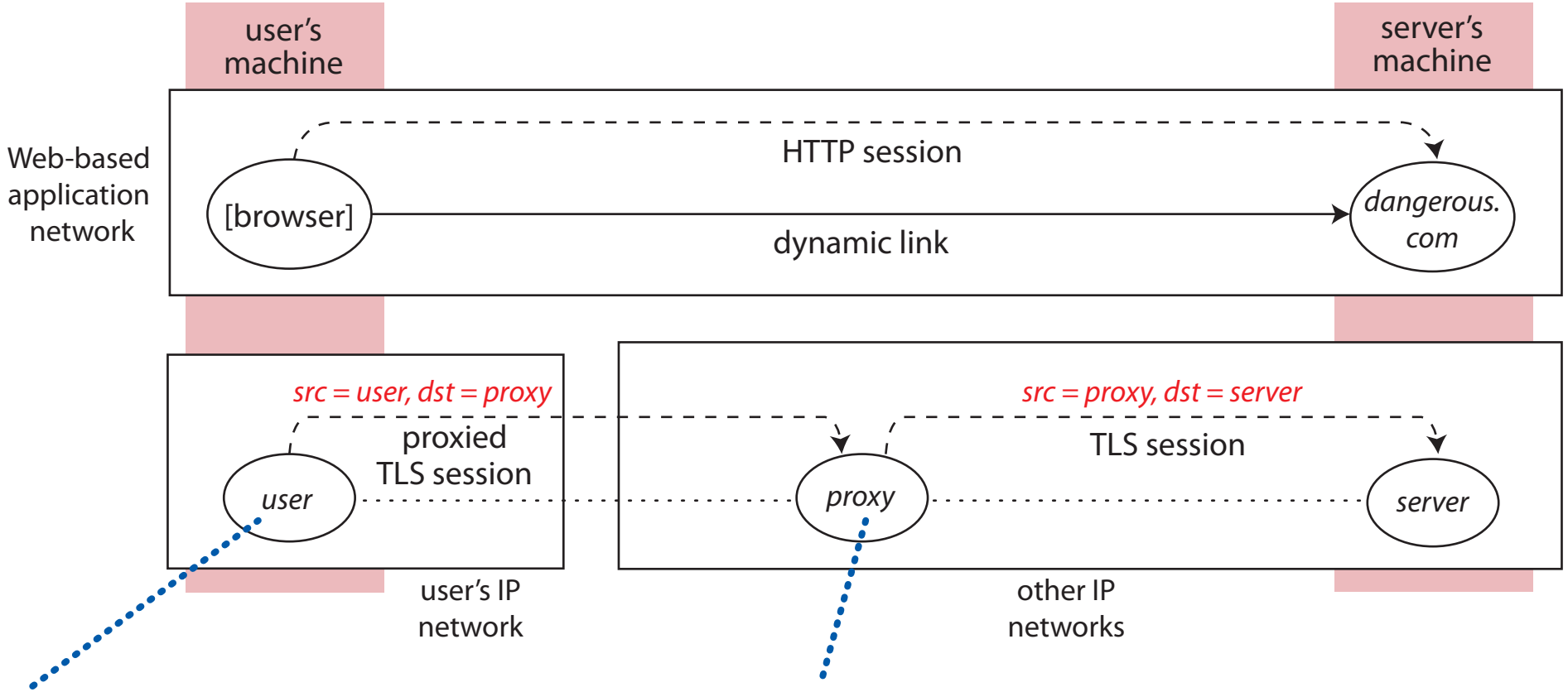
2. How are packets routed through the overlay to hide them from attackers?

3. What are the secrets by which the filters recognize overlay packets?

PROXIES SUBVERT FILTERING

it may be benign—set up your network this way so some users can escape an over-simplified firewall

it may be ethical—user may want privacy, freedom from censorship, anonymity from acceptor of session



this IP interface uses proxy as destination, ignores HTTP request

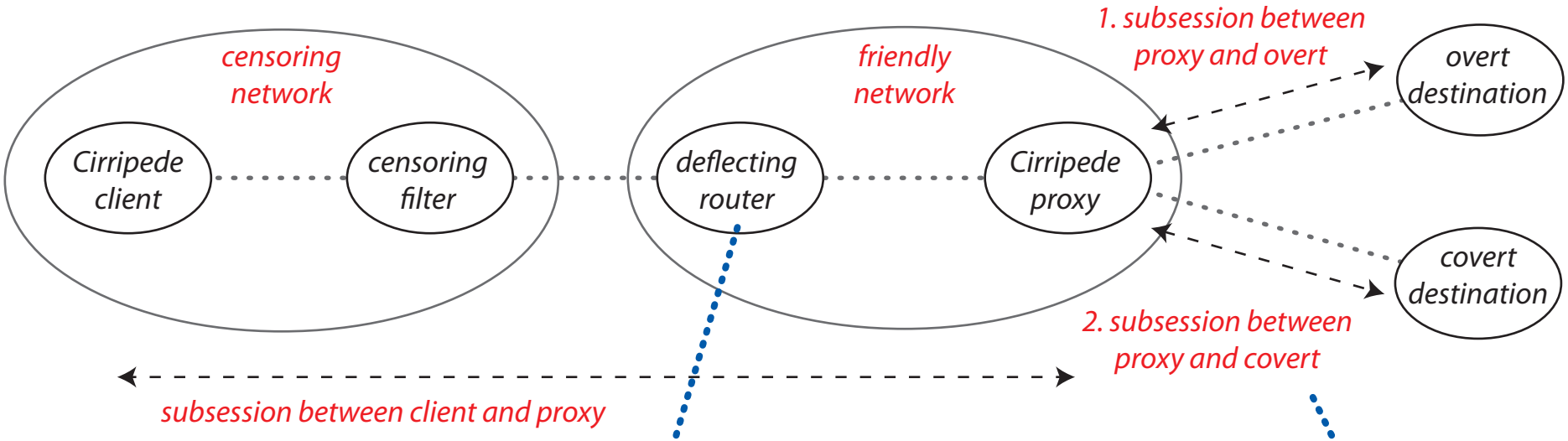
this IP interface looks up dangerous.com in DNS

PROXIES SUBVERT FILTERING

in the last scheme, if user is subject to censoring in access network, access network could block packets to known proxies

to use decoy routing, Cirripede, Telex, user must first send a message to the friendly network that, in a secret code, tells the friendly network that he wants to use the service

packet timing, order of TCP packets, unused packet fields, pseudo-random packet fields



over-rides normal routing for this client

requires a very complex session protocol

to get privacy from proxies, need Tor

* filtering resources

  - there is a vast tree of paths from sources to target (root of tree), across multiple networks
  - advantages of filtering near target: fewer paths to cover, fewer packets to process, target's network has the incentives to do it
  - advantages of filtering near sources: attack traffic is dropped sooner, source's access network knows more about the source, there are many more resources near sources than near the target
  - disadvantages of filtering near sources: lack of incentives, coordination is difficult (complex, hard to secure)

* today's practices

  - filter near targets, with virtualization in clouds to provide the resources necessary during attacks
  - also replicate and virtualize the protected target, so there is more capacity to withstand attacks
  - reduce amplification with SYN cookies, longer DNS caching, etc.

Compositional Intrastructure Security

* **interactions with bridging**
  - **if a network is isolated (no bridging), then attacker needs physical access**

* **interactions with layering**
  - **we must have this property for filtering to be valid: no packet is received on a link that was not sent on the link**
  - **sounds easy, but what if the link belongs to a cloud tenant, and is implemented in a network shared among tenants?**
  - **signature-based filtering looks for specific keywords in specific positions in a packet, so it had better know the exact layers above the filter (better to filter in each network separately)**

* **interactions with middleboxes**
  - **packet filters are middleboxes, so all the issues with encryption and middleboxes are relevant**
  - **the interaction between proxies and filtering is extremely important; there is a battle for control among users, proxy hosts, network AAs, services**

* **interactions with routing**
  - **routing might send packets of the same session along different routes because of failures or load-balancing, which conflicts with the need for session affinity**

* **interactions with session protocols**
  - **SYN cookies "dumb down" TCP, which is not helpful to those trying to extend it; SYN-flood-defense servers don't have this bad interaction**