Advanced Internet Architectures

Carlos Borrego Iglesias

Carlos.Borrego@uab.cat

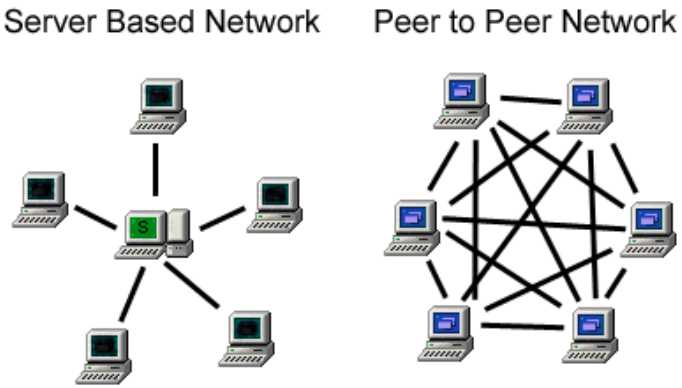
Departament d'Enginyeria de la Informació i de les Comunicacions
Universitat Autònoma de Barcelona

Tecnologies avançades d'Internet. Teaching material.

Contents

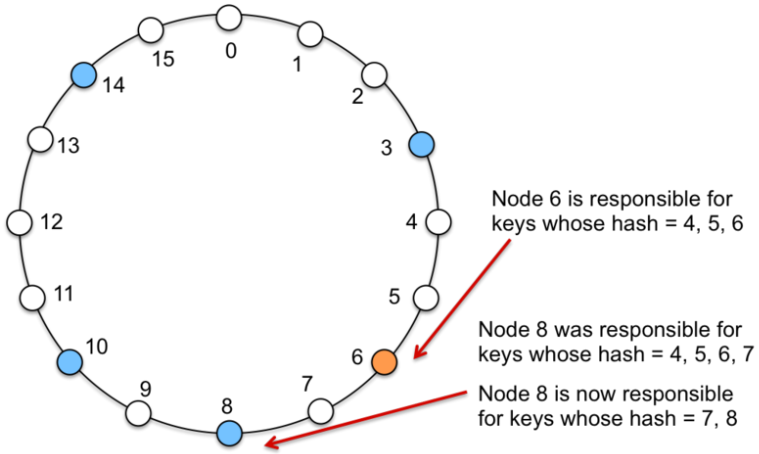
Contents

Contents



Peer-to-peer Architecture

- **Definition:** A peer-to-peer (abbreviated to P2P) computer network is one in which each computer in the network can act as a client or server for the other computers in the network (peers).
- **Types:**
  - **Pure peer-to-peer systems:** The entire network consists solely of equipotent peers.
  - **Centralized peer-to-peer systems:** A central server is used for indexing functions and to bootstrap the entire system.
  - **Hybrid peer-to-peer systems:** Allow such infrastructure nodes to exist, often called supernodes.



Distributed hash table

- **Definition:** decentralized distributed hash table system.
- **Transparency**: Any participating node can efficiently retrieve the value associated with a given key.
- **Good scalability:** A change in the set of participants causes a minimal amount of disruption.

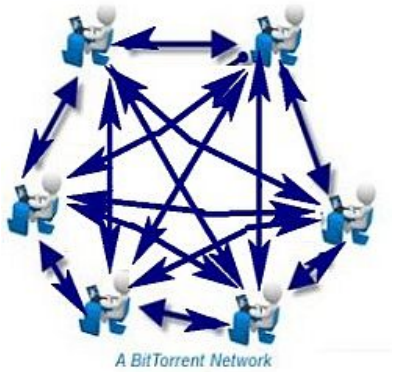
Distributed hash table

- Peers form a logical ring of 2^n potential peers.
- Peer location in the ring is calculated using a hash function of size n ($f(peer) = x / x \in [0 : 2^n - 1]$)
- Items (files for example), use the same hash function to be located in the ring.
- Peers take responsibility for $\sim$ 1/n of the keyspace (items located between the peer position and the position position of its next peer).

Distributed hash table

- Once you add a peer to the network, it finds a place on the ring to sit between two other peers, and takes responsibility for some of the keys in its sibling peers.
- When *uploading* an item to a network, the custodian of the item informs the item responsible.
- Peers interested in this item will contact the item responsible to retrieve the list of peers that custody the item.
- Peers that retrieve bits of an item inform the item responsible.

Contents



File sharing networks

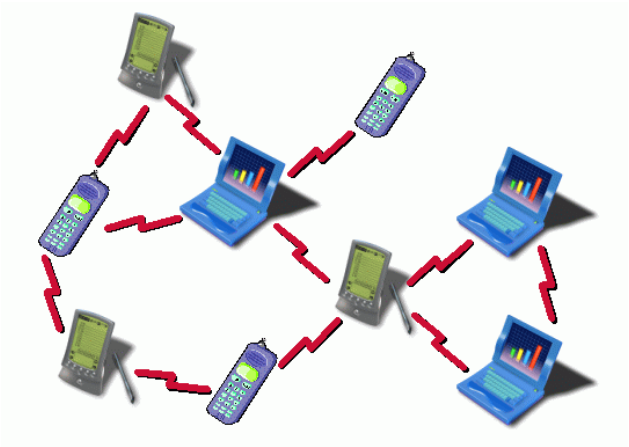
- **Definition:** Allows users to download media files using a P2P software client that searches for other connected computers.
- A user creates a small **torrent descriptor**. They then make the file itself available through a BitTorrent node acting as a **seed**. Those with the torrent descriptor file can give it to their own BitTorrent nodes which, acting as **peers** or **leechers**, download it by connecting to the seed and/or other peers.
- The file being distributed is divided into segments called **pieces**.
- **Examples: gnutella, bittorrent, G2 and the eDonkey**

Other Applications

- **Software publication and distribution:** Example: Linux, several games.
- **Streaming media:** P2PTV and PDTP. Applications include TVUPlayer, Joost, CoolStreaming, Cybersky-TV, PPLive, LiveStation, Giraffic and Didiom.
- **Spotify:** Peer-to-peer network along with streaming servers to stream music to its desktop music player.
- **Bitcoin:** Peer-to-peer based digital currency.
- **WWW Search:** For example: YaCy, a free distributed search engine, built on principles of peer-to-peer networks.
- **Skype:** one of the most widely used internet phone applications is using P2P technology.

Contents

Contents



Ad Hoc Networks

- **Definition:** A wireless ad hoc network is a decentralized type of wireless network. It does not rely on a preexisting infrastructure. Instead, each node participates in routing by forwarding data for other nodes.

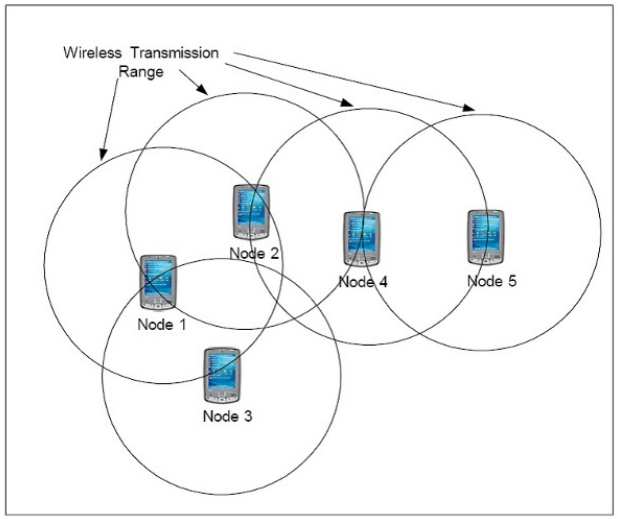
Addressing in AdHoc networks Options:

- **Zero configuration networking** Host/routers may automatically configure an interface with an IPv4 address within the 169.254/16 prefix that is valid for communication with other devices connected to the same physical (or logical) link.
- The address selection algorithm is based on computing a hash on the interface's MAC address. (RFC 3927).
- **The multicast Domain Name System** When an mDNS client needs to resolve a host name, it broadcasts a query message that asks the host having that name to identify itself. That target machine then multicasts a message that includes its IP address. All machines in that subnet use that information to update their mDNS caches. RFC 6762.

Routing in AdHoc networks

- **Pro-active:** Maintains fresh lists of destinations and their routes by periodically distributing routing tables throughout the network. Examples: B.A.T.M.A.N., Optimized Link State Routing Protocol (RFC 3626).
- **Reactive:** Finds a route on demand by flooding the network with Route Request packets. Examples: Ad hoc On-demand Distance Vector(AODV) (RFC 3561), Dynamic Source Routing (RFC 4728).
- **Hybrid (both pro-active and reactive):** Combines the advantages of proactive and of reactive routing. The routing is initially established with some proactively prospected routes and then serves the demand from additionally activated nodes through reactive flooding. The choice for one or the other method requires predetermination for typical cases. Examples: ZRP (Zone Routing Protocol).

Contents



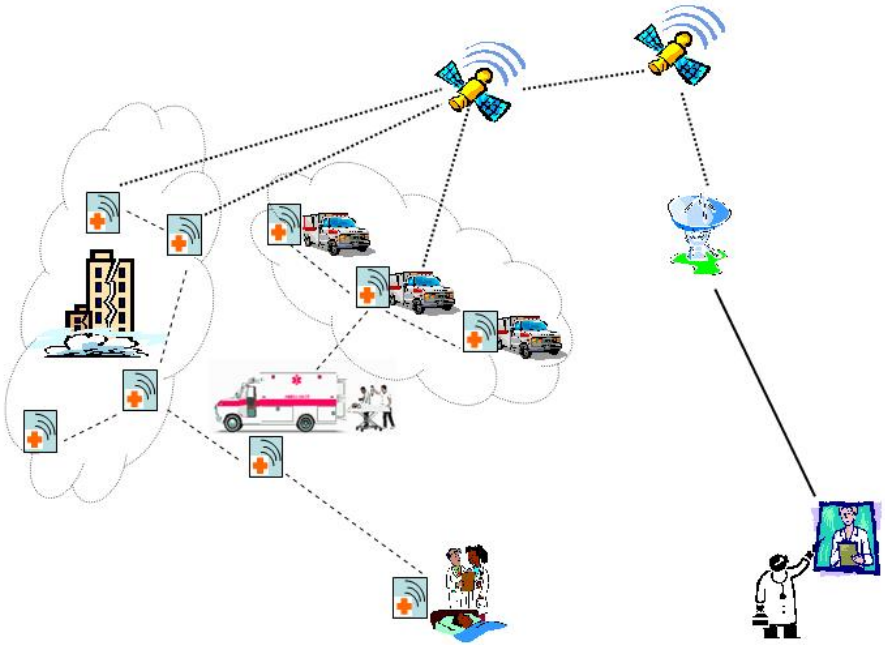
Mobile Ad Hoc Networks (MANETs)

- **Definition:** A mobile ad hoc network (MANET) is a self-configuring infrastructureless network of mobile devices connected by wireless.
- Each device in a MANET is free to move independently in any direction, and will therefore change its links to other devices frequently. Each must forward traffic unrelated to its own use, and therefore be a router.

Types

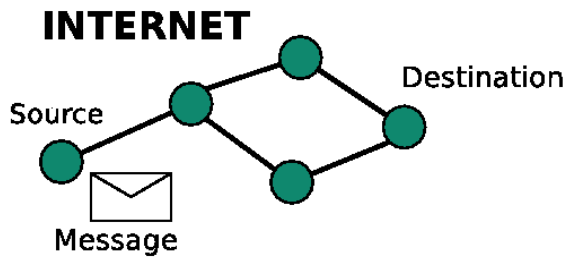
- **Vehicular Ad hoc Networks (VANETs):** Used for communication among vehicles and between vehicles and roadside equipment
- **Smart phone ad hoc networks (SPANs):** leverage the existing hardware (primarily Bluetooth and Wi-Fi) in smart phones to create peer-to-peer networks without relying on traditional network infrastructure. SPANs differ from traditional hub and spoke networks, such as Wi-Fi Direct, in that they support multi-hop relays.
- **Internet based mobile ad hoc networks (iMANET):** Ad hoc networks that link mobile nodes and fixed Internet-gateway nodes. For example, multiple sub-MANETs may be connected in a classic Hub-Spoke VPN to create a geographically distributed MANET.

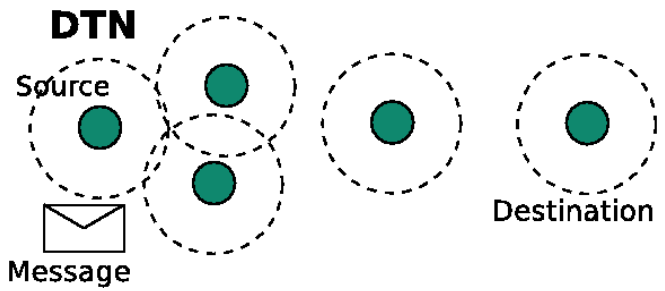
Contents



Delay-tolerant networking (DTN)

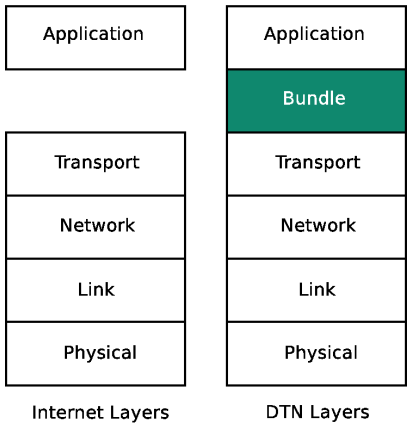
- **Definition:** Delay-tolerant networking (DTN) is an approach to computer network architecture that seeks to address the technical issues in heterogeneous networks that may lack continuous network connectivity.
- Follow the **Store-carry-and-forward** paradigm: Information is sent to an intermediate station where it is kept and sent at a later time to the final destination or to another intermediate station.





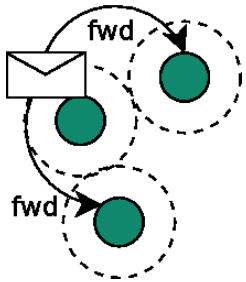
Delay and Disruption Tolerant Networks (DTN)

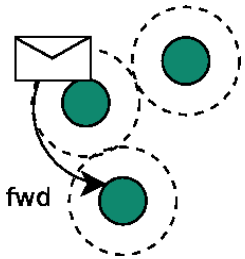
- **Research:** DTN, opportunistic networks, challenged networks, partitioned networks...
- **Characteristics:** Intermittent connectivity, asymmetric bandwidths, long and variable latency, ambiguous mobility patterns...
- New applications made possible
- **Applications:** WSN, space applications, emergency scenarios, medical application in developing countries...



Delay and Disruption Tolerant Networking Architecture

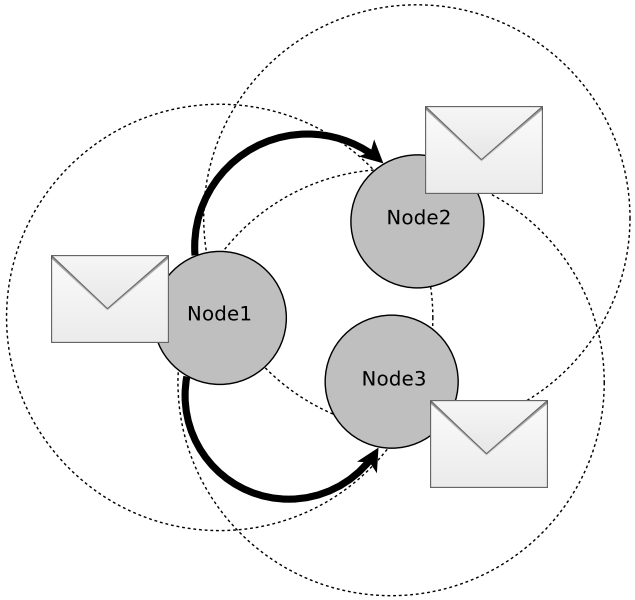
- **RFC4838:** Describes an architecture for DTN using *Store-carry-and-forward*.
- **RFC5050:** Describes the end-to-end protocol, block formats, an abstract service description for the exchange of messages (bundles) in DTN





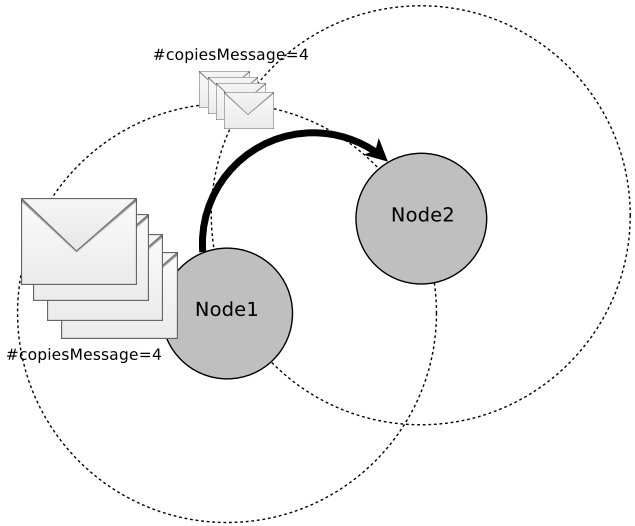
Routing in Delay and Disruption Tolerant Networks

- **Definition:** next hop decision process
- Complex decision due to DTN characteristics
- **Classification:** based on the number of copies, geographical, probabilistic, epidemic...
- Routing code is located in the intermediate nodes
- Lack of proposals for scenarios with different applications with different routing needs



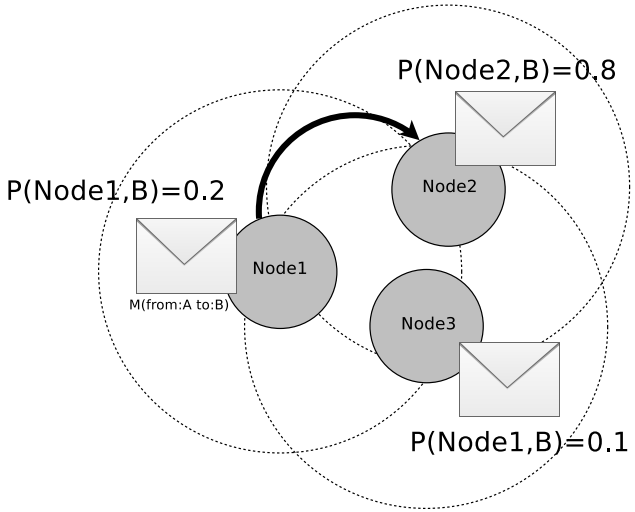
Epidemic Routing

- Epidemic routing is flooding-based in nature, as nodes continuously replicate and transmit messages to newly discovered contacts that do not already possess a copy of the message.



Spray And Wait Routing

- When a new message is created in the system, a number L is attached to that message indicating the maximum allowable copies of the message in the network.
- During the **spray phase**, the source of the message is responsible for "spraying", or delivery, one copy to L distinct "relays".
- When a relay receives the copy it simply holds the message until the destination is encountered directly (**wait phase**).



Probabilistic Routing Protocol (Prophet Routing)

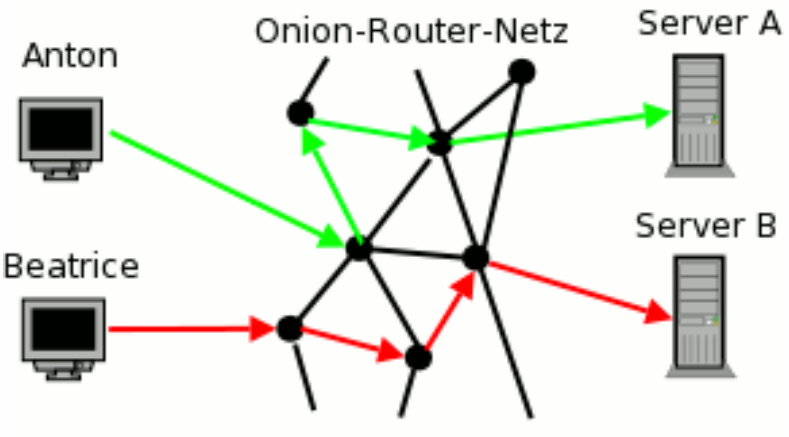
- The Probabilistic Routing Protocol maintains a set of probabilities for successful delivery to known destinations in the DTN.
- It replicates messages during opportunistic encounters only if the Mule that does not have the message appears to have a better chance of delivering it.

Probabilistic Routing Protocol (Prophet Routing)

- When M encounters E, the predictability for E is increased: $P(M,E)new = P(M,E)old + (1 - P(M,E)old) * L_{encounter}$
- $L_{encounter}$ is an initialisation constant.
- The predictabilities for all destinations D other than E are 'aged':
- $P(M,D)new = P(M,D)old * \gamma^K$
- γ is the aging constant and K is the number of time units that has elapsed since the last aging.
- Predictabilities are exchanged between M and E and the 'transitive' property of predictability is used to update the predictability of destinations D for which E has a P(E,D) value on the assumption that M is likely to meet E again:
- $P(M,D)new = P(M,D)old + (1 - P(M,D)old) * P(M,E) * P(E,D) * \beta$
- β is a scaling constant.

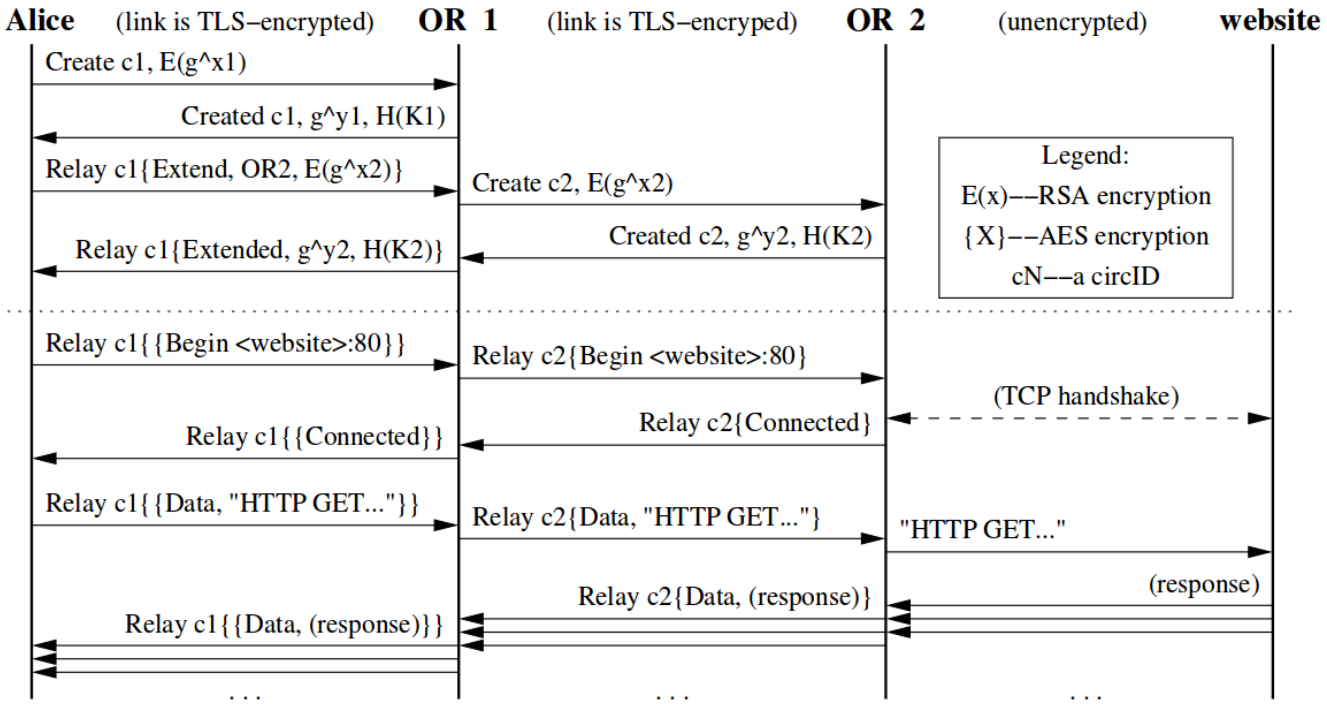
Contents

Contents

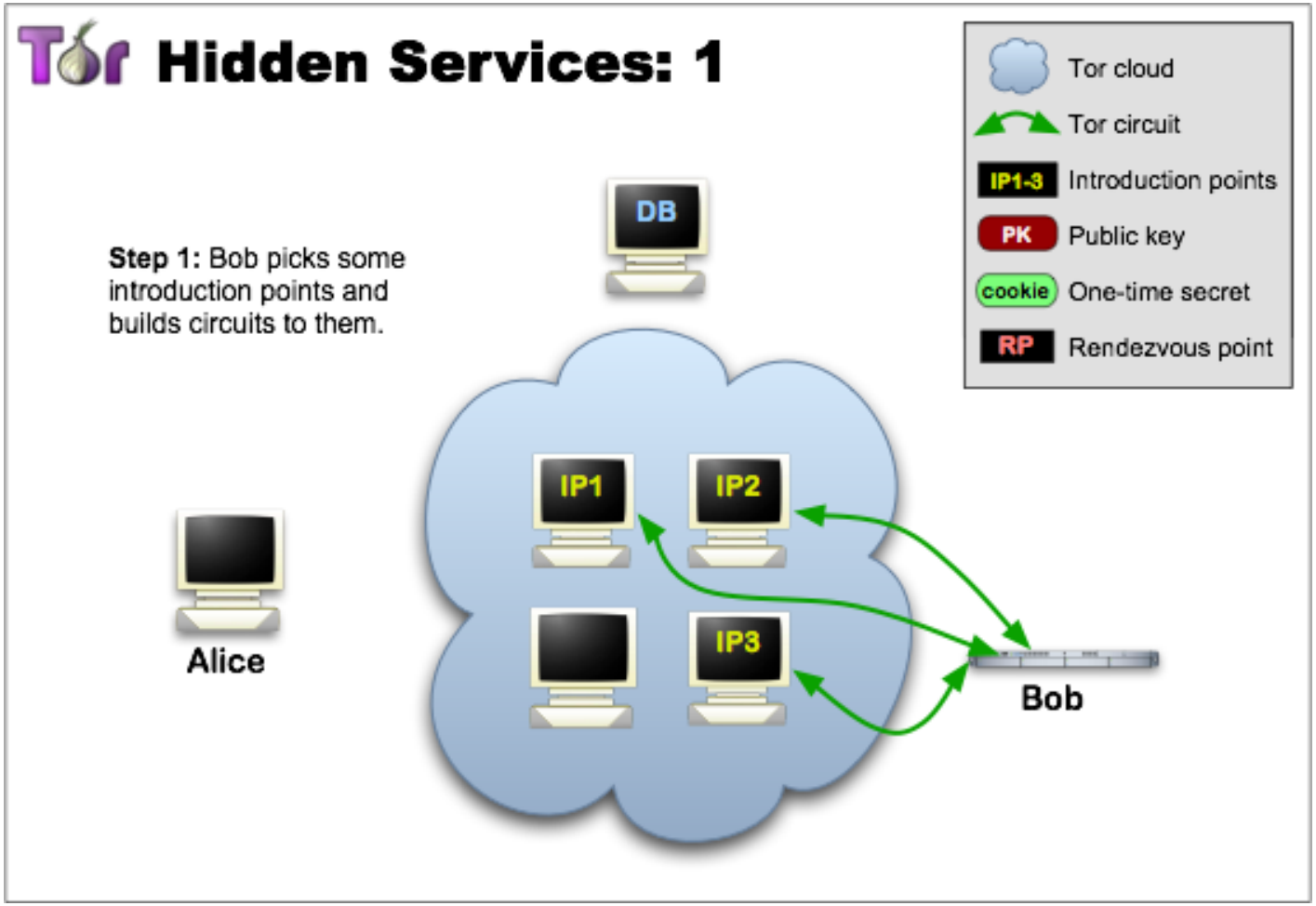


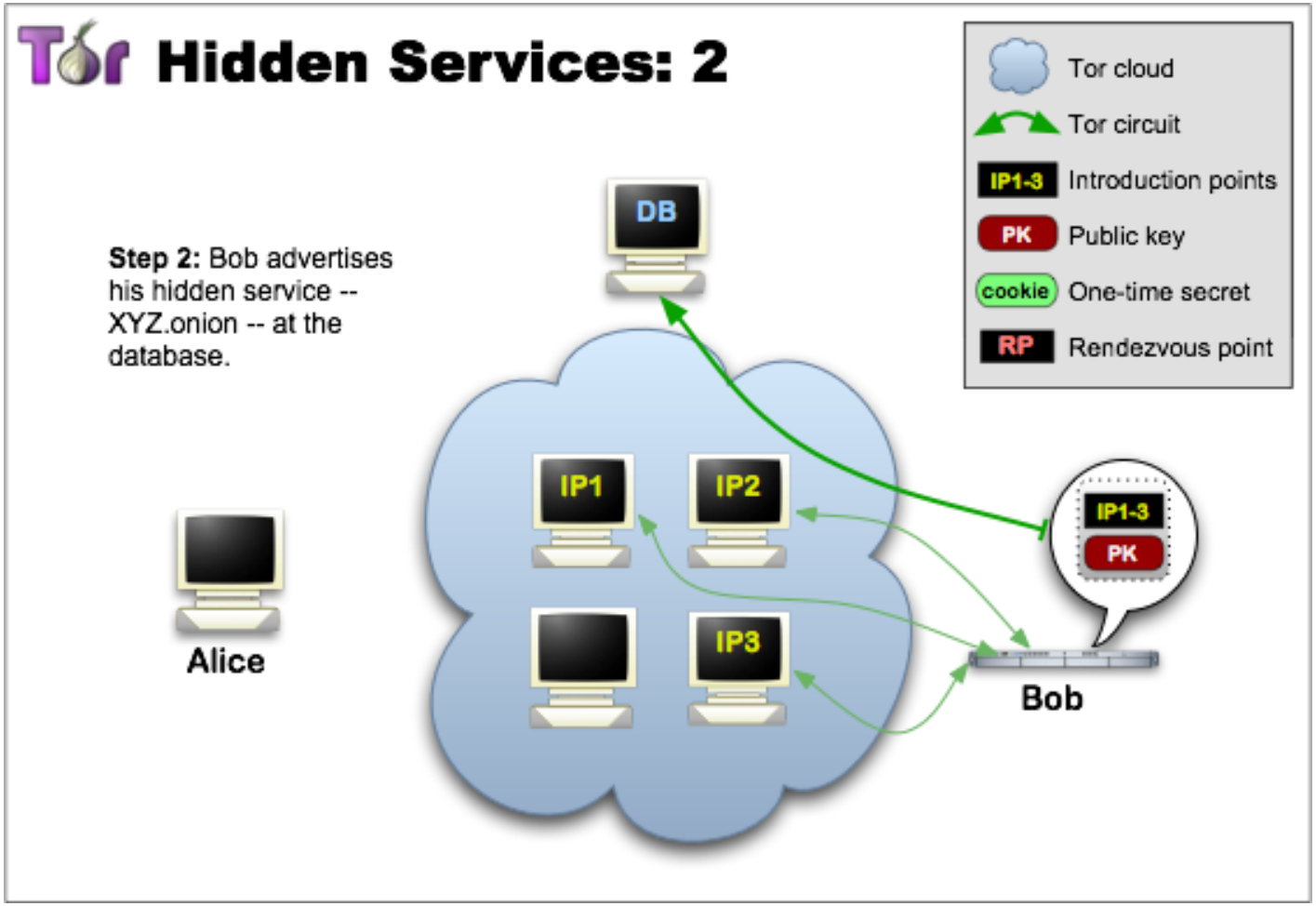
TOR, the Onion Router

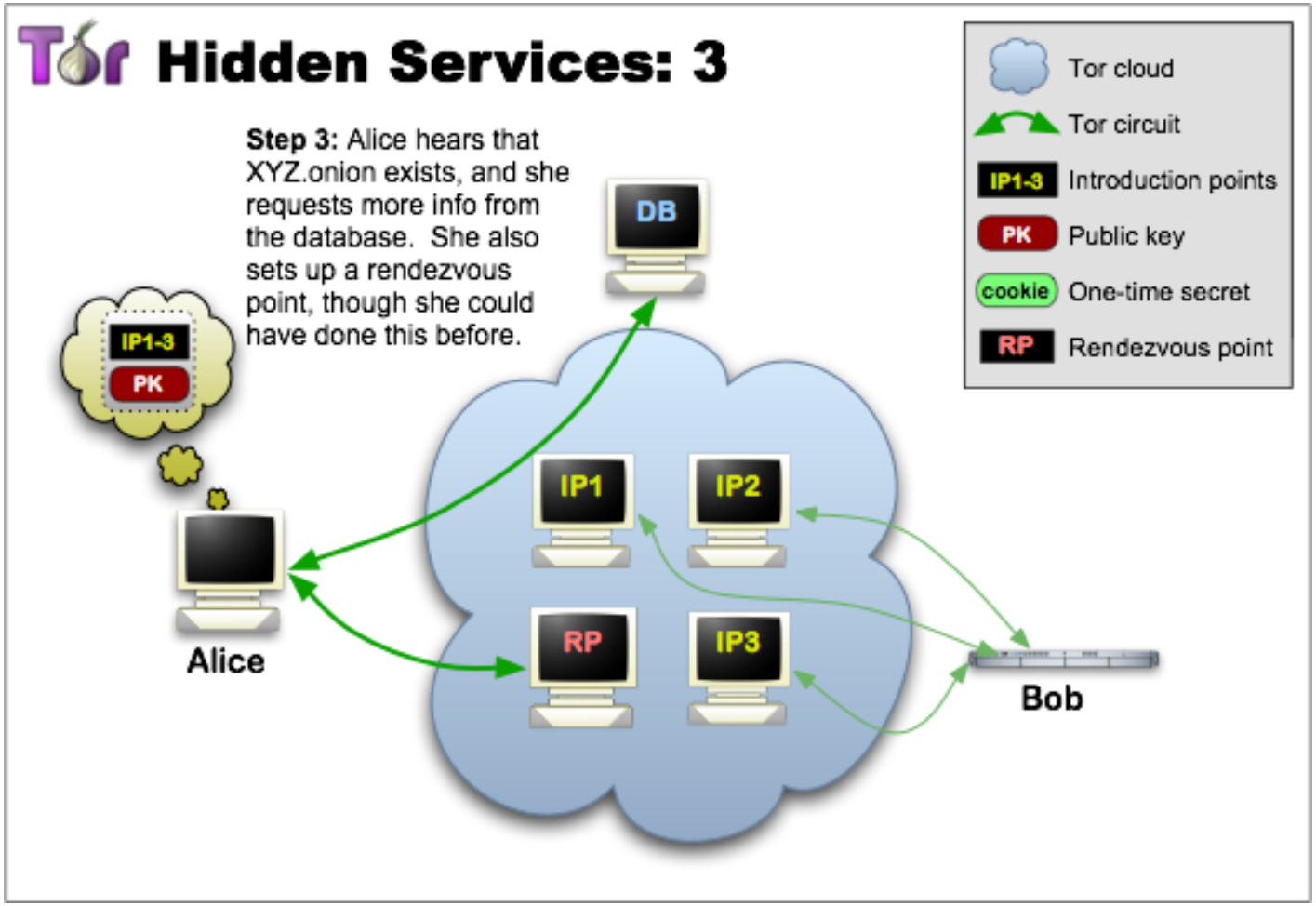
- **Definition:** System intended to enable online anonymity.
- **Onion Routing** refers to the layered nature of the encryption service: The original data are encrypted and re-encrypted multiple times, then sent through successive Tor relays, each one of which decrypts a "layer"of encryption before passing the data on to the next relay and, ultimately, its destination.

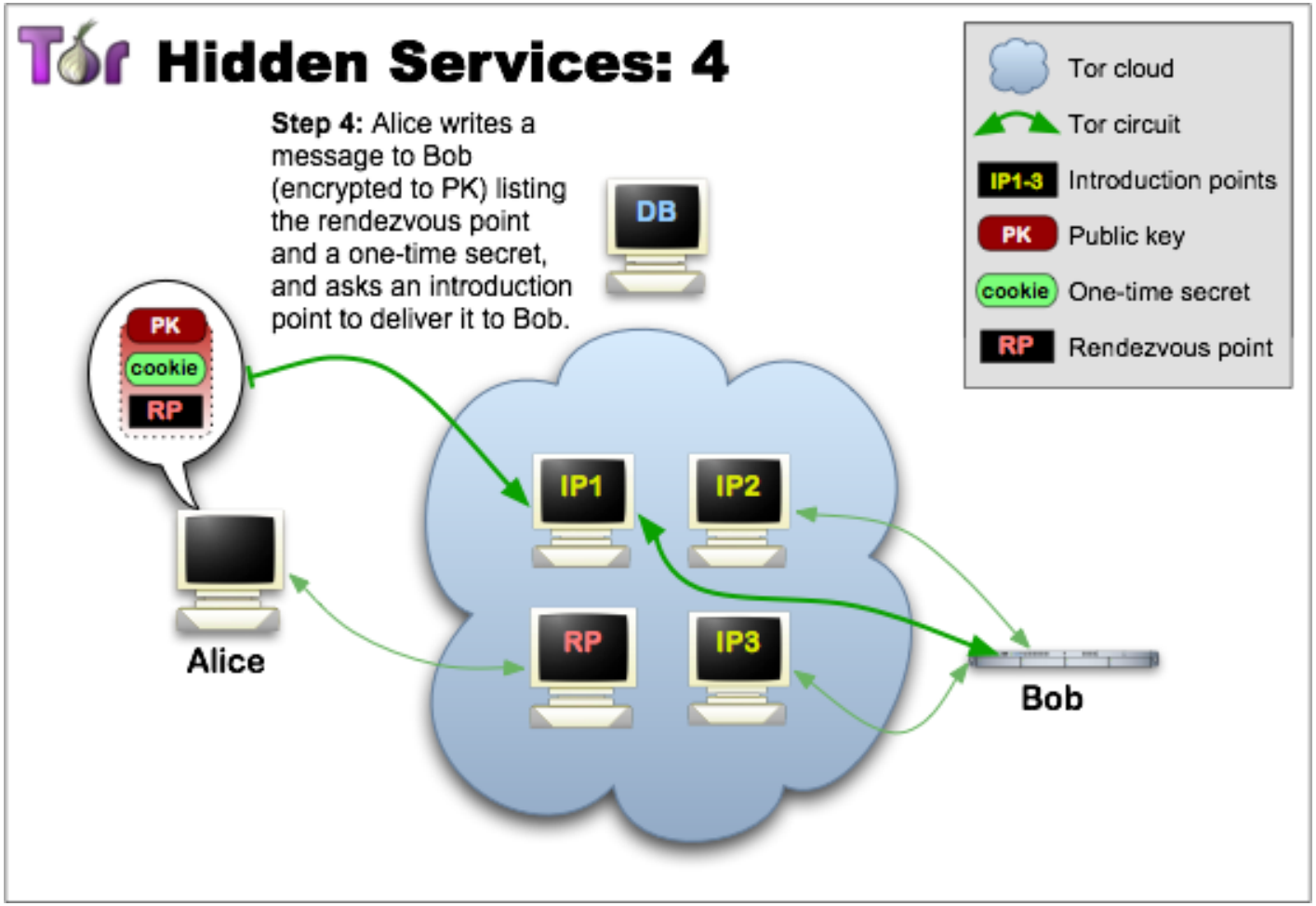

Alice
(link is TLS–encrypted)
OR 1
(link is TLS–encryped)
OR 2
(unencrypted)
website
Create c1, E(g^x1)
Created c1, g^y1, H(K1)
Relay c1{Extend, OR2, E(g^x2)}
Create c2, E(g^x2)
Created c2, g^y2, H(K2)
Relay c1{Extended, g^y2, H(K2)}
Legend:
E(x)––RSA encryption
{X}––AES encryption
cN––a circID
Relay c1{{Begin <website>:80}}
Relay c2{Begin <website>:80}
(TCP handshake)
Relay c2{Connected}
Relay c1{{Connected}}
Relay c1{{Data, "HTTP GET..."}}
Relay c2{Data, "HTTP GET..."}
"HTTP GET..."
(response)
Relay c2{Data, (response)}
Relay c1{{Data, (response)}}

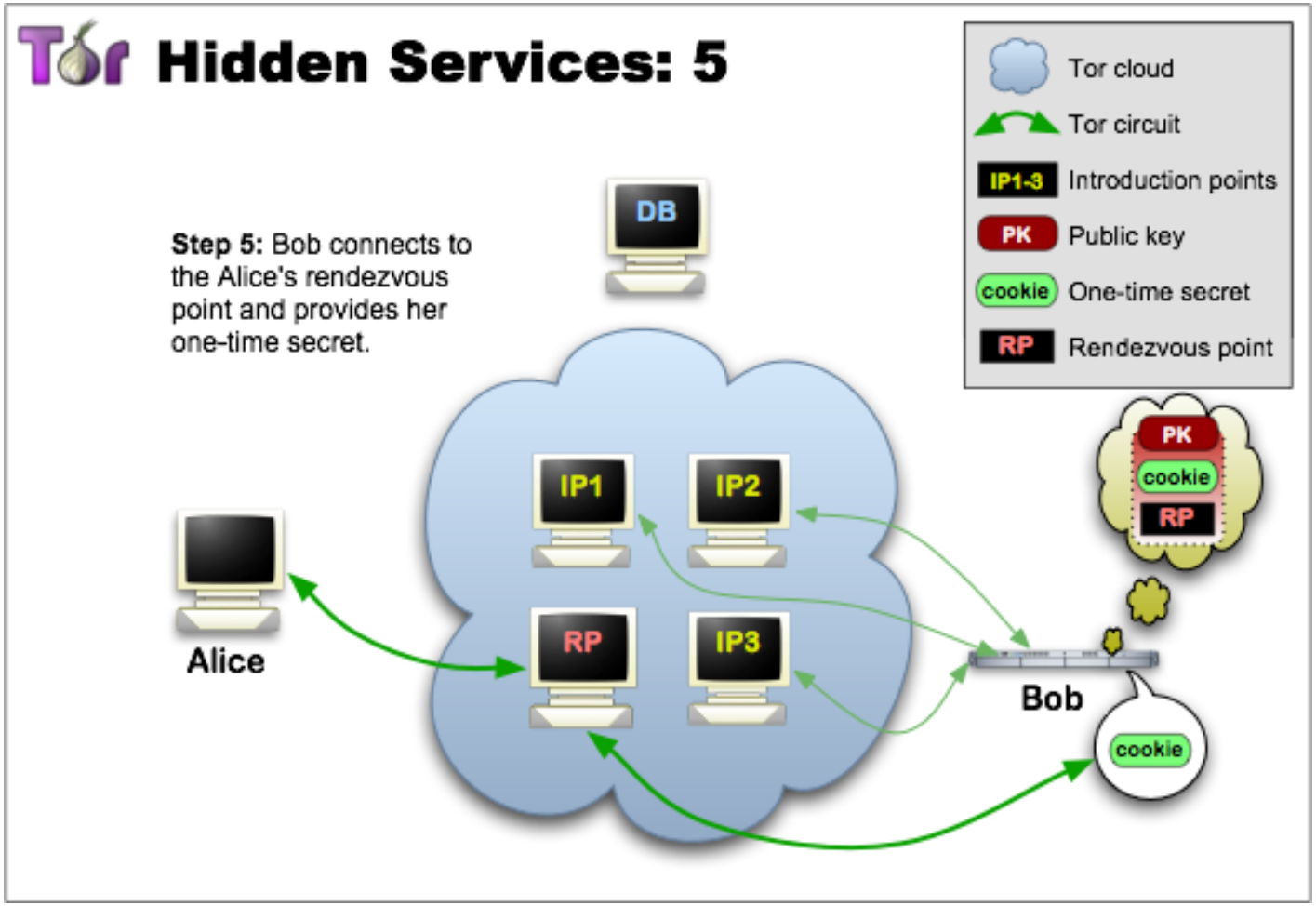
Contents

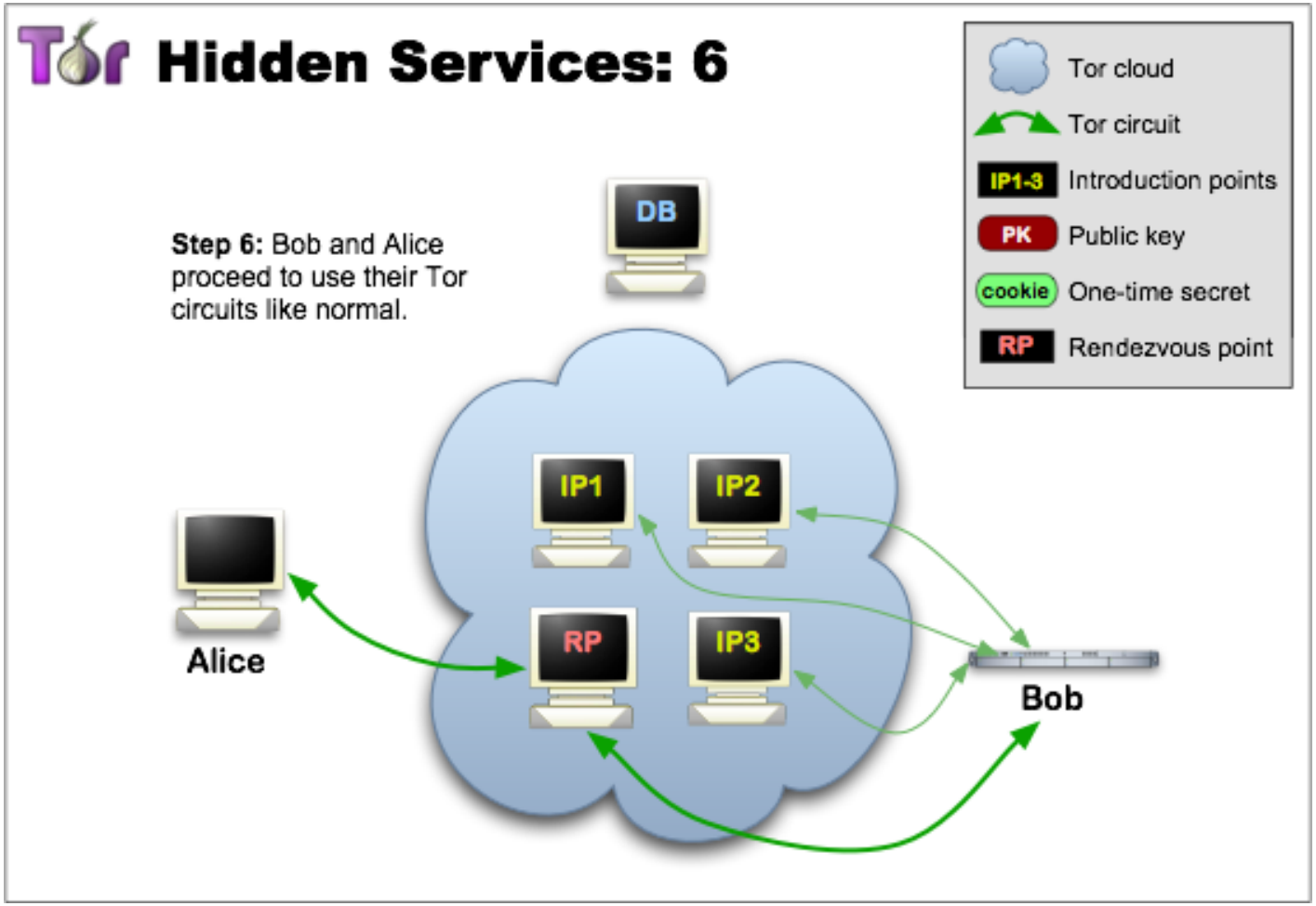

Hidden Services: 1
Step 1: Bob picks some introduction points and builds circuits to them.
DB
Tor cloud
Tor circuit
IP1-3
Introduction points
PK
Public key
cookie
One-time secret
RP
Rendezvous point
IP1
IP2
IP3
Alice
Bob


Hidden Services: 2
Step 2: Bob advertises his hidden service -- XYZ.onion -- at the database.
Tor cloud
Tor circuit
IP1-3 Introduction points
PK Public key
cookie One-time secret
RP Rendezvous point
DB
IP1
IP2
IP3
IP1-3
PK
Alice
Bob


Hidden Services: 3
Step 3: Alice hears that XYZ.onion exists, and she requests more info from the database. She also sets up a rendezvous point, though she could have done this before.
Tor cloud
Tor circuit
IP1-3 Introduction points
PK Public key
cookie One-time secret
RP Rendezvous point
DB
IP1-3
PK
IP1
IP2
RP
IP3
Alice
Bob


Hidden Services: 4
Step 4: Alice writes a message to Bob (encrypted to PK) listing the rendezvous point and a one-time secret, and asks an introduction point to deliver it to Bob.
DB
PK
cookie
RP
Alice
IP1
IP2
RP
IP3
Bob
Tor cloud
Tor circuit
IP1-3
Introduction points
PK
Public key
cookie
One-time secret
RP
Rendezvous point


Hidden Services: 5
Step 5: Bob connects to the Alice's rendezvous point and provides her one-time secret.
Tor cloud
Tor circuit
IP1-3 Introduction points
PK Public key
cookie One-time secret
RP Rendezvous point
DB
IP1
IP2
RP
IP3
PK
cookie
RP
Alice
Bob
cookie


Hidden Services: 6
Step 6: Bob and Alice proceed to use their Tor circuits like normal.
Tor cloud
Tor circuit
IP1-3 Introduction points
PK Public key
cookie One-time secret
RP Rendezvous point
DB
IP1
IP2
RP
IP3
Alice
Bob