Kerberos

What is Kerberos?

- Network authentication protocol
- Developed at MIT in the mid 1980s
- Available as open source or in supported commercial software

Why Kerberos?

- Sending usernames and passwords in the clear breech in the security of the network.
- Each time a password is sent in the clear, there is a chance for interception.

Firewall vs. Kerberos?

- Firewalls make a risky assumption: that attackers are coming from the outside. In reality, attacks frequently come from within.
- Kerberos assumes that network connections (rather than servers and work stations) are the weak link in network security.

Design Requirements

- Interactions between hosts and clients should be encrypted.
- Must be convenient for users (or they won't use it).
- Protect against intercepted credentials.

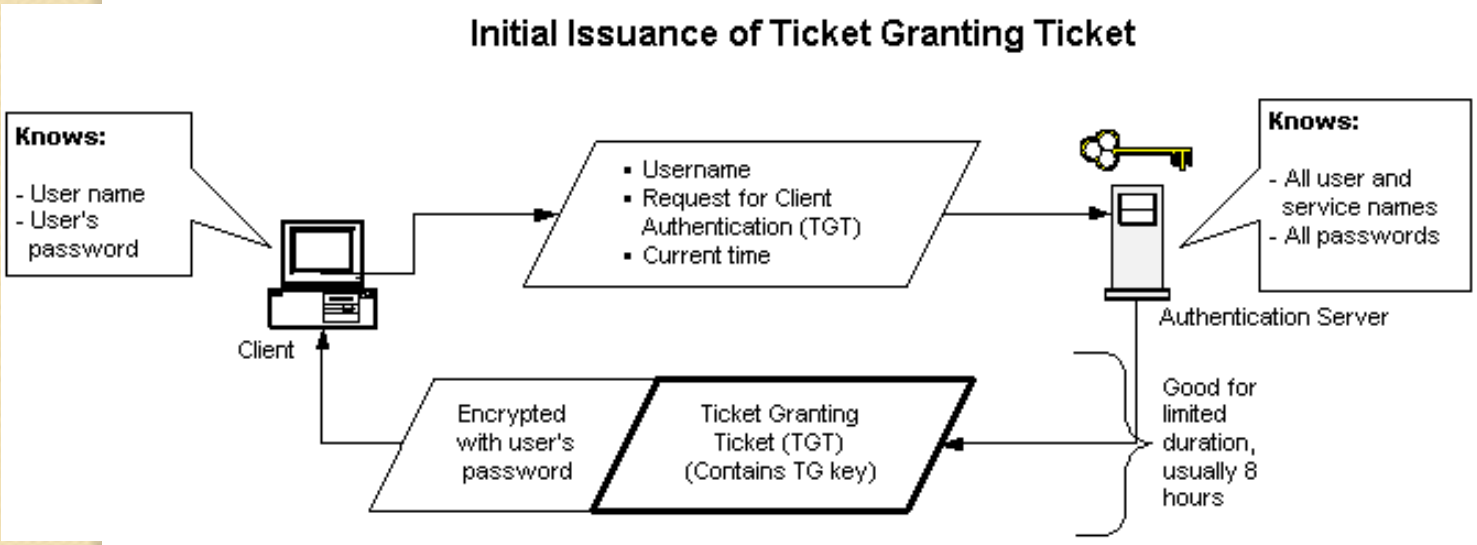
Cryptography Approach

- Private Key: Each party uses the same secret key to encode and decode messages.
- Uses a trusted third party which can vouch for the identity of both parties in a transaction. Security of third party is imperative.

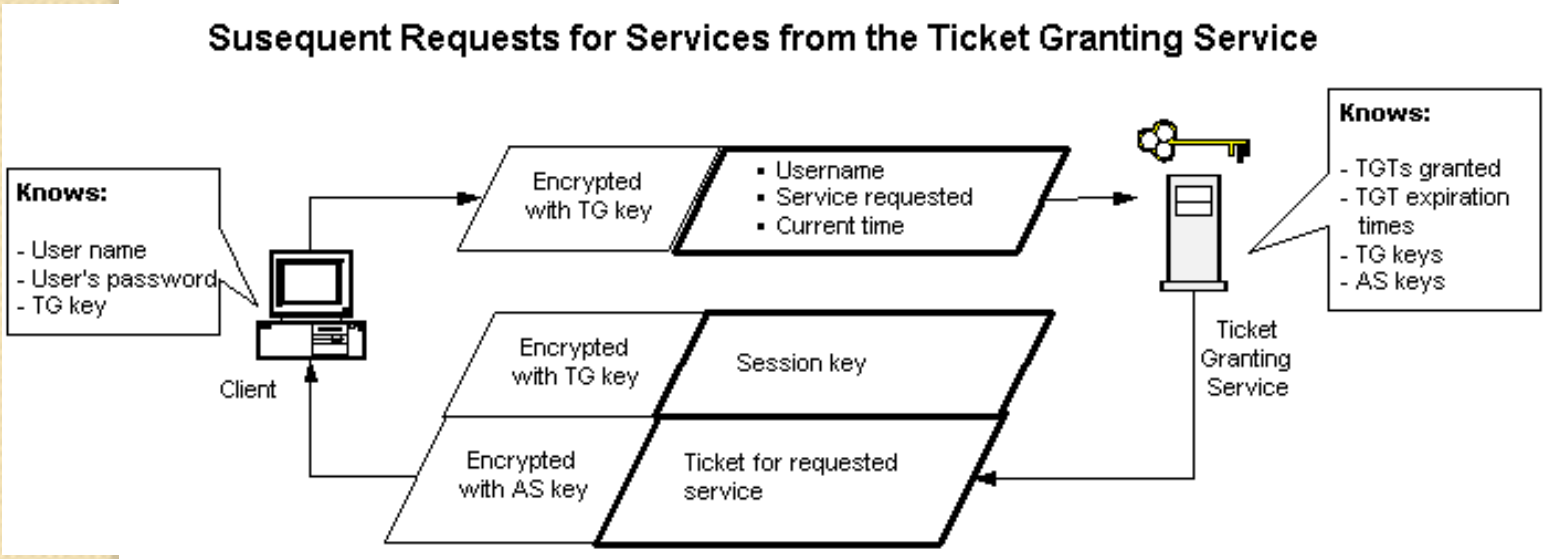
How does Kerberos work?

- Instead of client sending password to application server:
  - Request **Ticket** from authentication server
  - Ticket and encrypted request sent to application server
- How to request tickets without repeatedly sending credentials?
  - **Ticket granting ticket (TGT)**

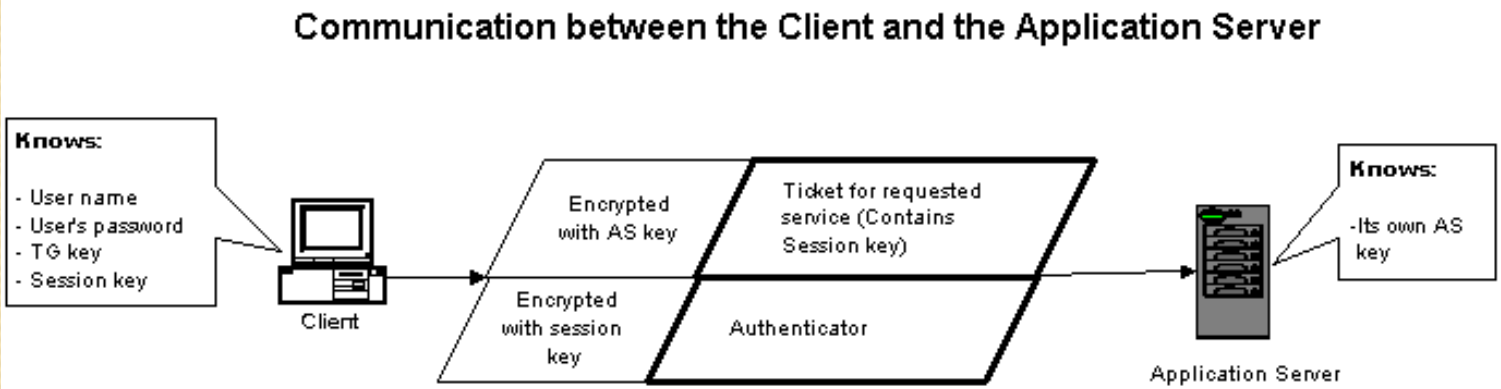
How does Kerberos work?: Ticket Granting Tickets



How does Kerberos Work?: The Ticket Granting Service



How does Kerberos work?: The Application Server



Applications

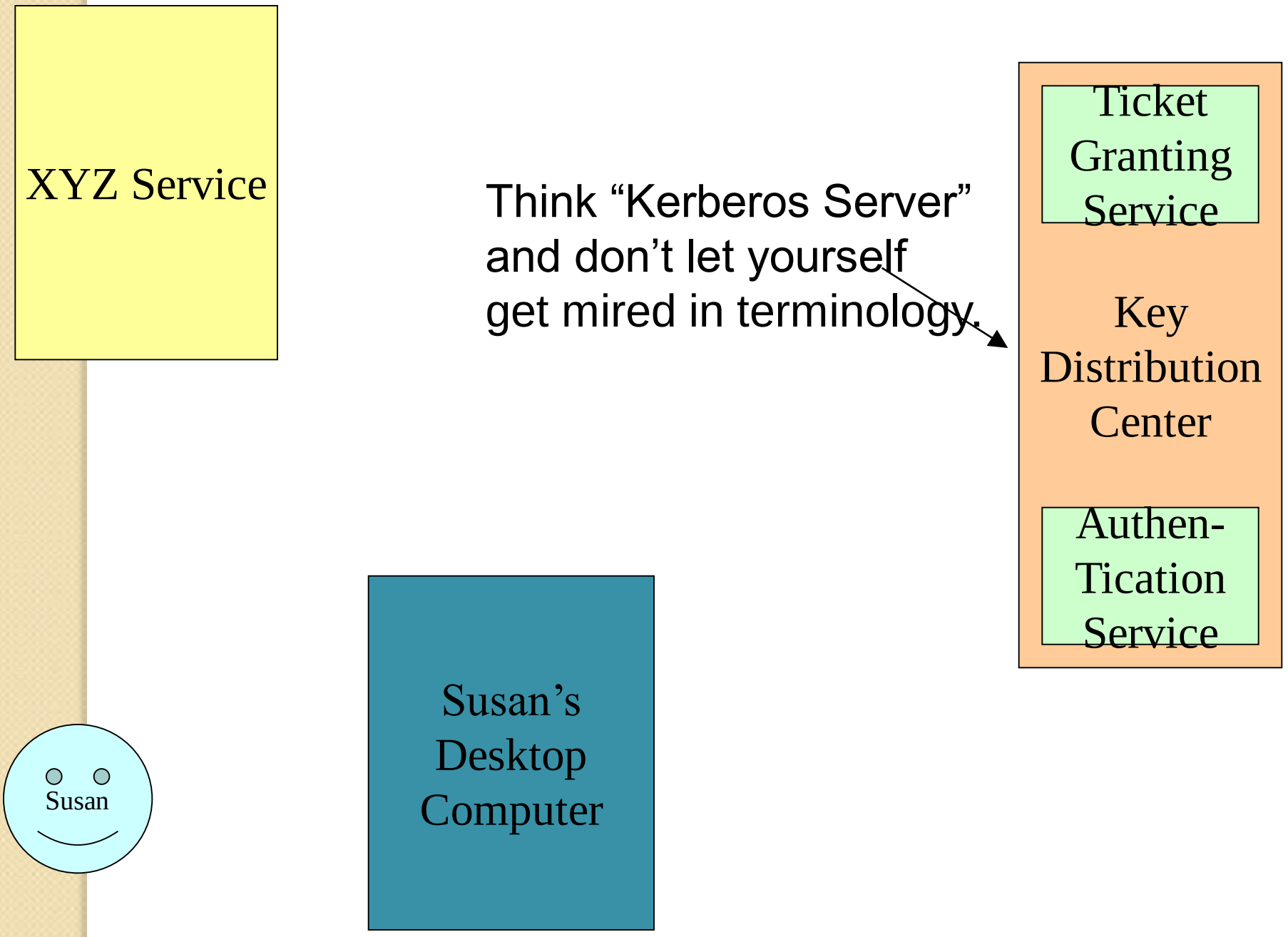
- Authentication
- Authorization
- Confidentiality
- Within networks and small sets of networks

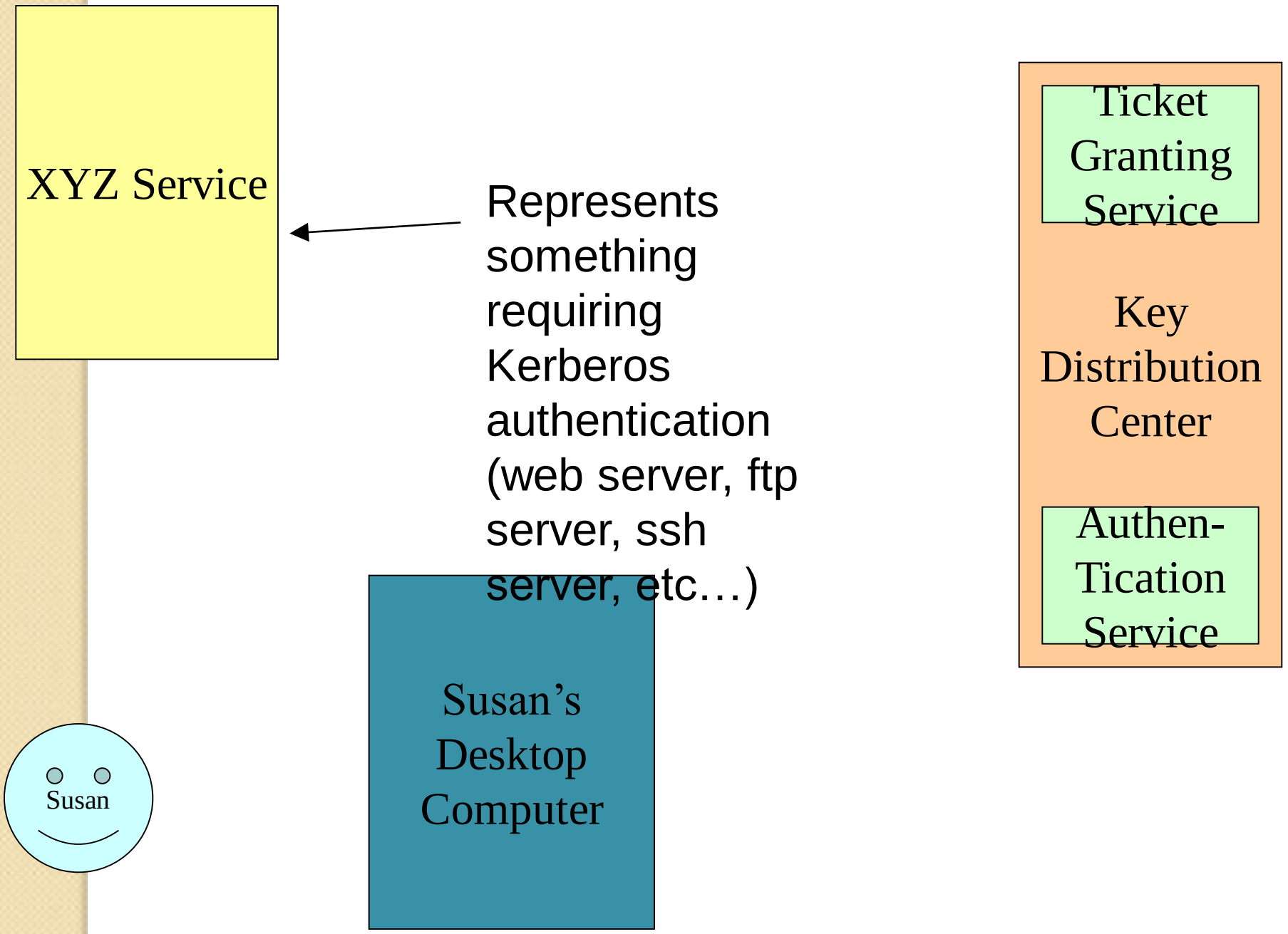
Weaknesses and Solutions

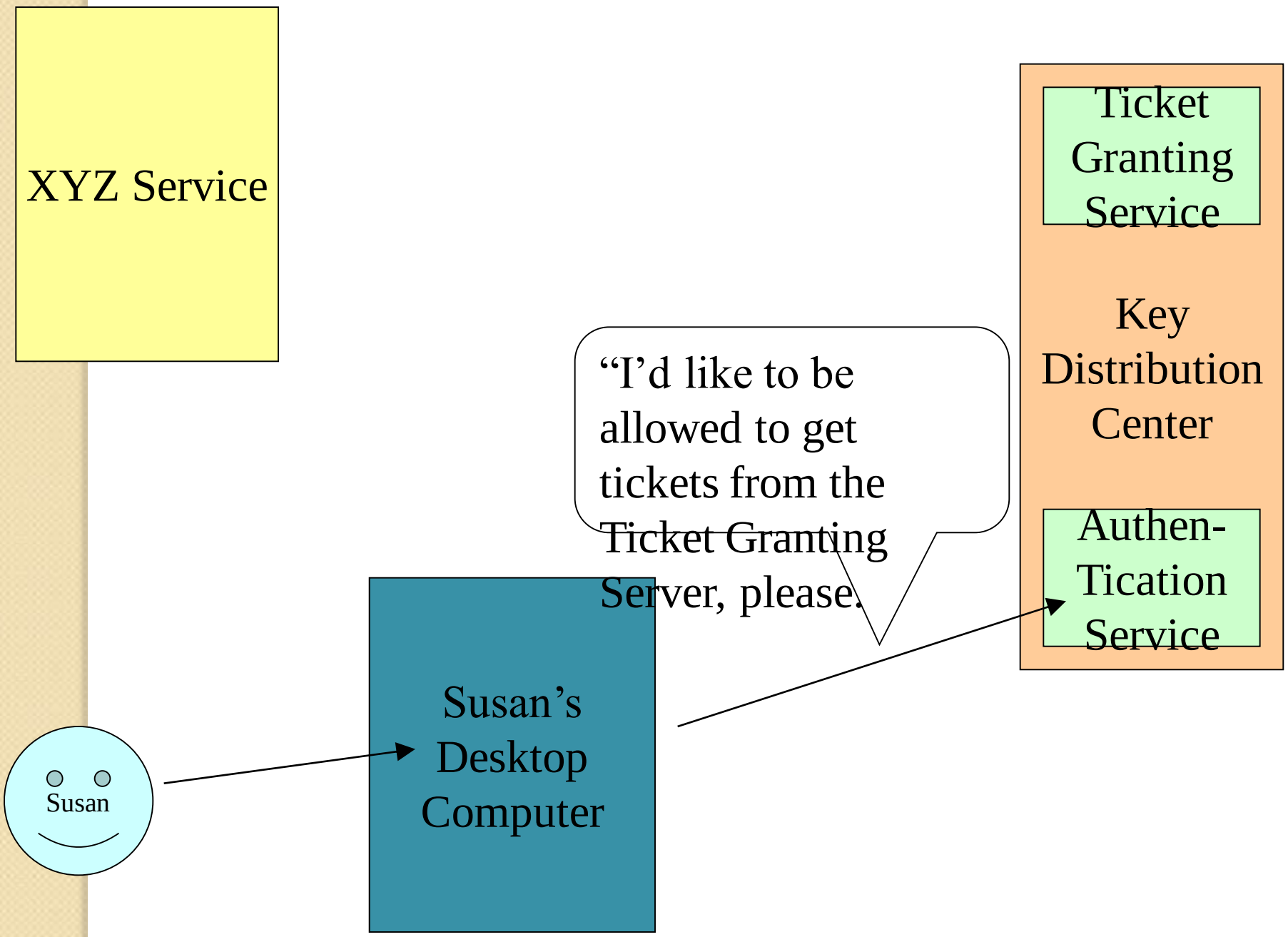
If TGT stolen, can be used to access network services.	Only a problem until ticket expires in a few hours.
Subject to dictionary attack.	Timestamps require hacker to guess in 5 minutes.
Very bad if Authentication Server compromised.	Physical protection for the server.

The Competition: SSL

SSL	Kerberos
Uses public key encryption	Uses private key encryption
Is certificate based (asynchronous)	Relies on a trusted third party (synchronous)
Ideal for the WWW	Ideal for networked environments
Key revocation requires Revocation Server to keep track of bad certificates	Key revocation can be accomplished by disabling a user at the Authentication Server
Certificates sit on a users hard drive (even if they are encrypted) where they are subject to being cracked.	Passwords reside in users' minds where they are usually not subject to secret attack.
Uses patented material, so the service is not free. Netscape has a profit motive in wide acceptance of the standard.	Kerberos has always been open source and freely available.


XYZ Service
Think “Kerberos Server” and don’t let yourself get mired in terminology.
Ticket Granting Service
Key Distribution Center
Authen-Tication Service
Susan’s Desktop Computer
Susan


XYZ Service
Represents something requiring Kerberos authentication (web server, ftp server, ssh server, etc…)
Ticket Granting Service
Key Distribution Center
Authen-Tication Service
Susan's Desktop Computer
Susan


XYZ Service
Ticket Granting Service
Key Distribution Center
Authen-Tication Service
"I'd like to be allowed to get tickets from the Ticket Granting Server, please."
Susan's Desktop Computer
Susan

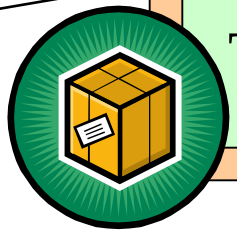
XYZ Service

Ticket Granting Service

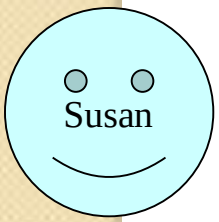
Key Distribution Center

Authen-Tication Service

"Okay. I locked this box with your secret password. If you can unlock it, you can use its contents to access my Ticket Granting Service."



Susan's Desktop Computer



XYZ Service

Ticket Granting Service

Key Distribution Center

Authen-Tication Service

Susan's Desktop Computer

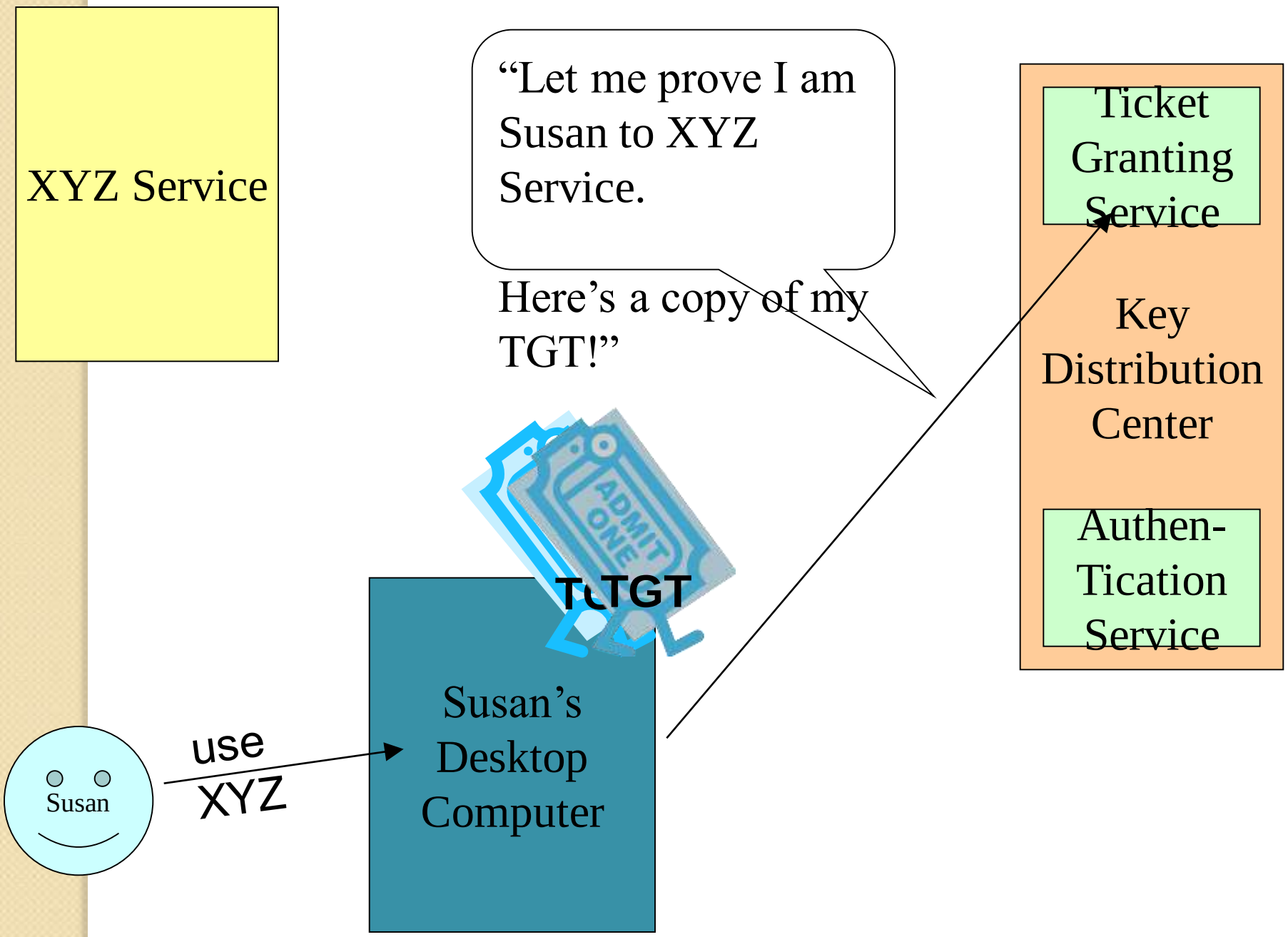
myPassword

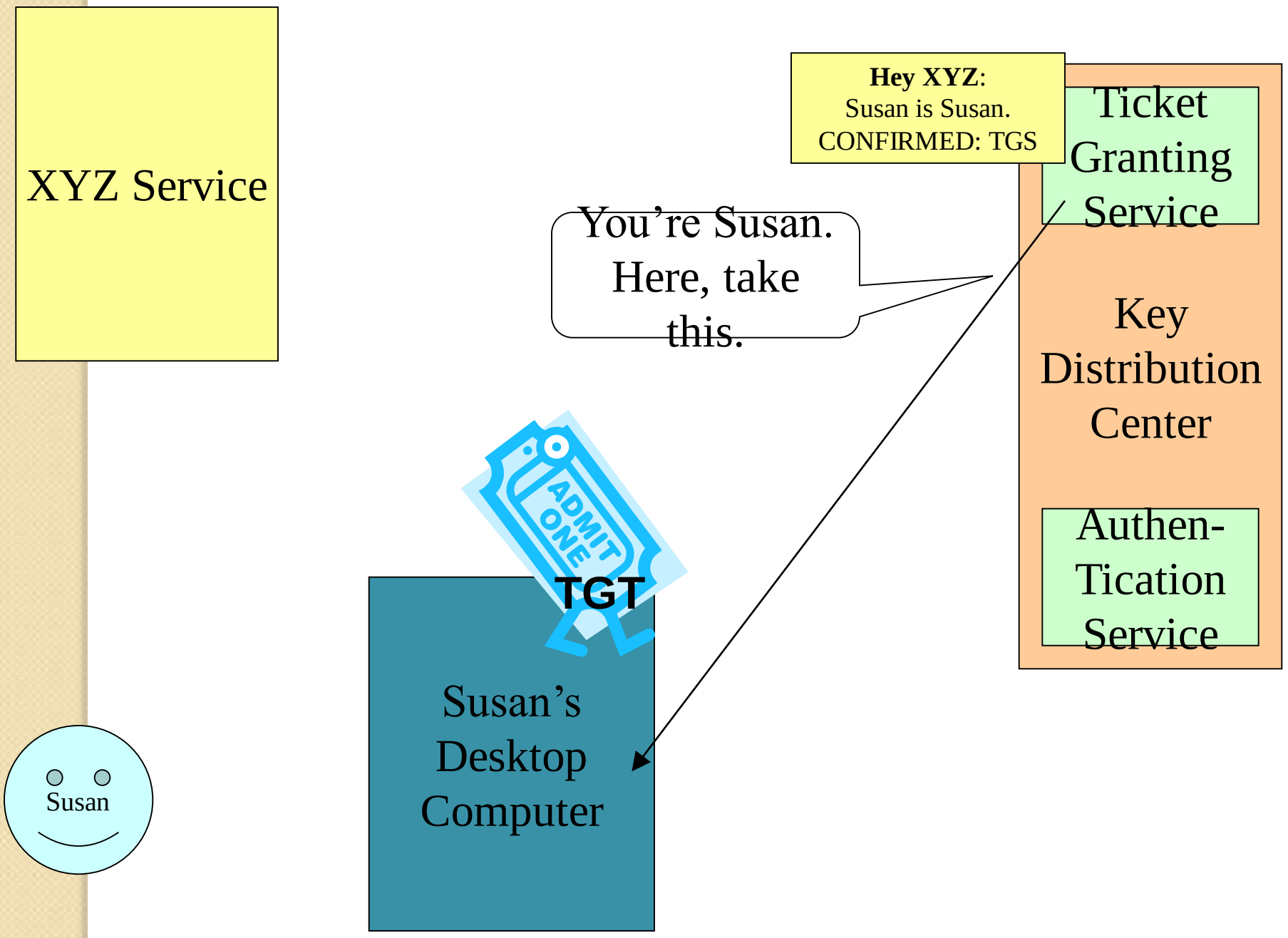
Susan

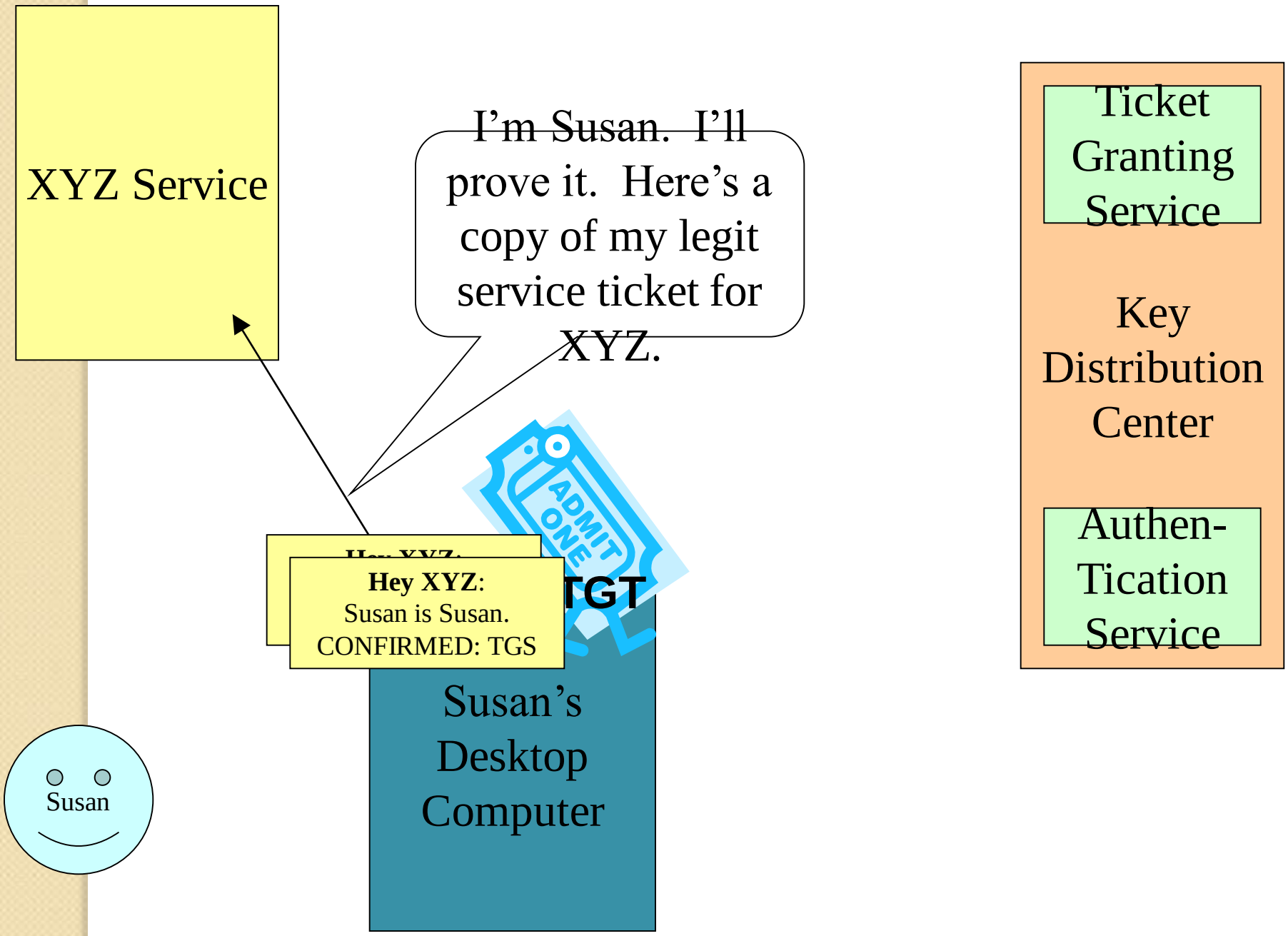
Because Susan was able to open the box (decrypt a message) from the Authentication Service, she is now the owner of a shiny “Ticket-Granting Ticket”.

The Ticket-Granting Ticket (TGT) must be presented to the Ticket Granting Service in order to acquire “service tickets” for use with services requiring Kerberos authentication.

The TGT contains no password information.


XYZ Service
"Let me prove I am Susan to XYZ Service.
Here's a copy of my TGT!"
Ticket Granting Service
Key Distribution Center
Authen-Tication Service
ADMIT ONE
TGT
Susan's Desktop Computer
Susan
use XYZ


XYZ Service
Hey XYZ:
Susan is Susan.
CONFIRMED: TGS
Ticket Granting Service
You're Susan. Here, take this.
Key Distribution Center
Authen-Tication Service
ADMIT ONE
TGT
Susan's Desktop Computer
Susan


XYZ Service
I'm Susan. I'll prove it. Here's a copy of my legit service ticket for XYZ.
Ticket Granting Service
Key Distribution Center
Authen-Tication Service
ADMIT ONE
TGT
Hey XYZ:
Susan is Susan.
CONFIRMED: TGS
Susan's Desktop Computer
Susan

XYZ Service

That's Susan alright. Let me determine if she is **authorized** to use me.

Hey XYZ:
Susan is Susan.
CONFIRMED: TGS

Ticket Granting Service

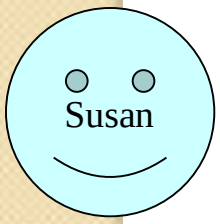
Key Distribution Center

Authen-Tication Service

TGT

Hey XYZ:
Susan is Susan.
CONFIRMED: TGS

Susan's Desktop Computer



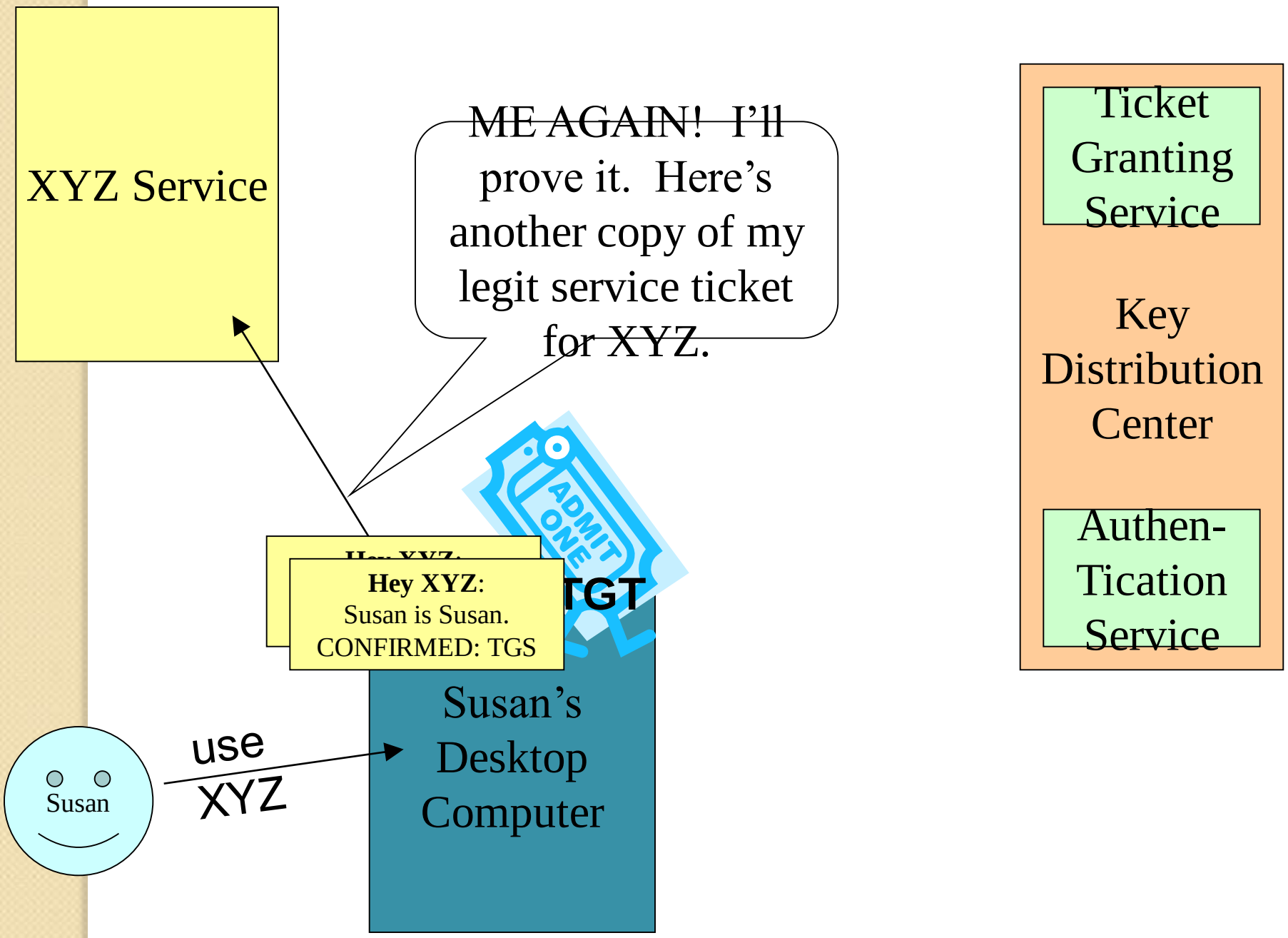
Authorization checks are performed by the XYZ service…

Just because Susan has **authenticated** herself does not inherently mean she is **authorized** to make use of the XYZ service.

One remaining note:

Tickets (your TGT as well as service-specific tickets) have expiration dates configured by your local system administrator(s). An expired ticket is unusable.

Until a ticket's expiration, it may be used repeatedly.


XYZ Service
ME AGAIN! I'll prove it. Here's another copy of my legit service ticket for XYZ.
Ticket Granting Service
Key Distribution Center
Authen-Tication Service
ADMIT ONE
TGT
Hey XYZ:
Susan is Susan.
CONFIRMED: TGS
Susan's Desktop Computer
Susan
use XYZ

XYZ Service

That's Susan… again. Let me determine if she is **authorized** to use me.

Hey XYZ:
Susan is Susan.
CONFIRMED: TGS

Ticket Granting Service

Key Distribution Center

Authen-Tication Service

TGT

Hey XYZ:
Susan is Susan.
CONFIRMED: TGS

Susan's Desktop Computer