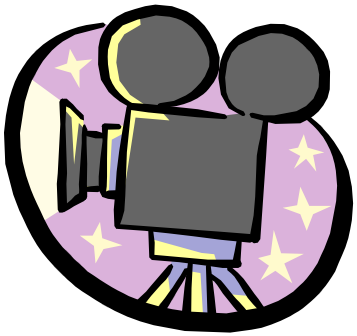
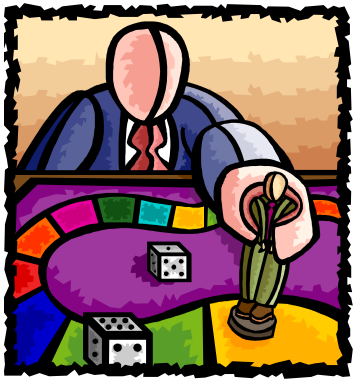
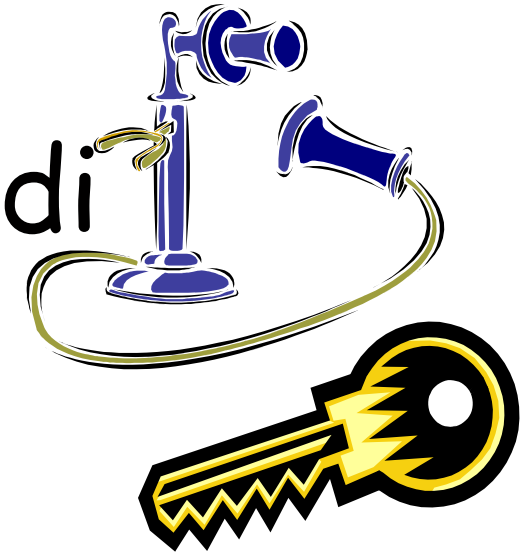
Multicast Sicuro e Scalabile in Presenza di Stati aggiornabili

Introduzione

- Comunicazione di gruppo sicura
  - Pay-per-view video streaming
  - Video On Demand (VOD)
  - Secure video conferencing
  - Online games

Comunicazione di Gruppo sicura



- Autorizzazione
- Multicast Sicuro
- Forward confidentiality (revoca)
- Backward confidentiality

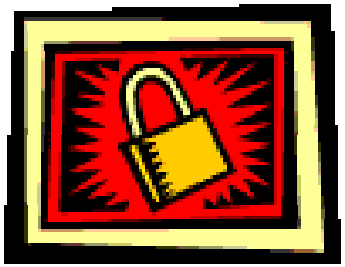
Tipi di Multicast Sicuro

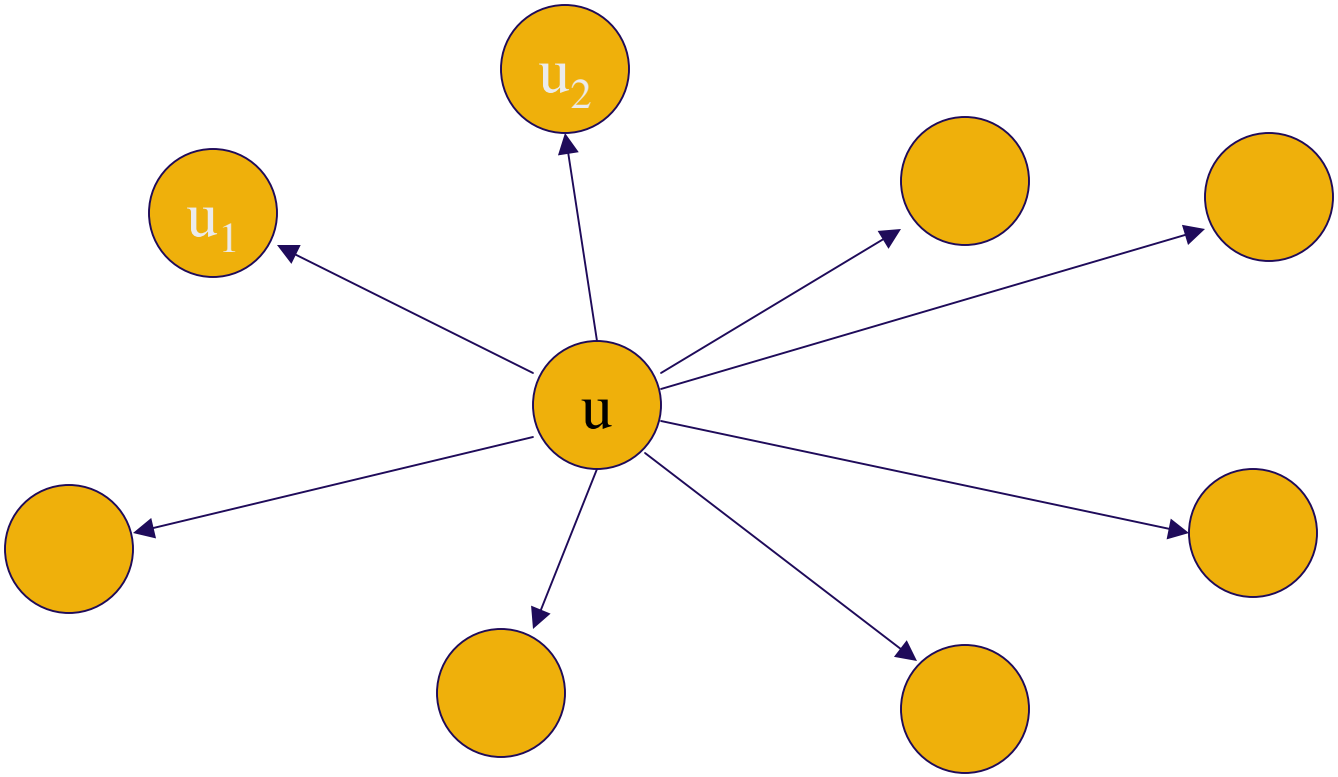
- Schemi Flat (piatti)
  - Esiste un solo layer fisico nel dominio
  - Esiste una sola chiave di encryption per i dati inviati in multicast da un singolo gruppo
  - La trasmissione costa poco, ma il re-keying costa molto
- Schemi Non-flat
  - Il dominio è fisicamente suddiviso in layers diversi
  - In genere, occorre ritrasmettere i dati
  - La trasmissione costa leggermente di più, ma il re-keying costa un pò meno.

Schemi di Multicast Sicuro

- Centralizzato
  - LKH, LKH+, OFT, …
- De-centralizzato
  - CBT, Iolus, Kronos, IGKMP
- Completamente distribuito
  - Burmester and Desmedt, Group Diffie-Hellman

Multicast di Gruppo Sicuro

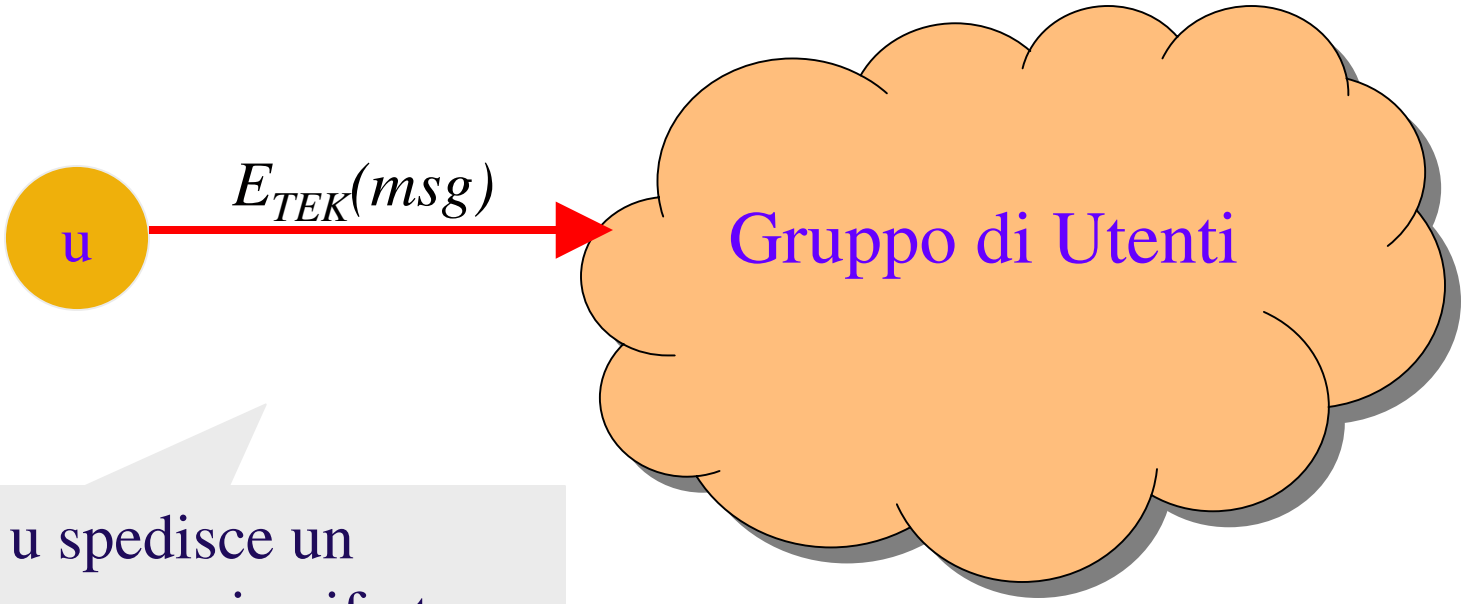




Schemi Flat: Assunzioni

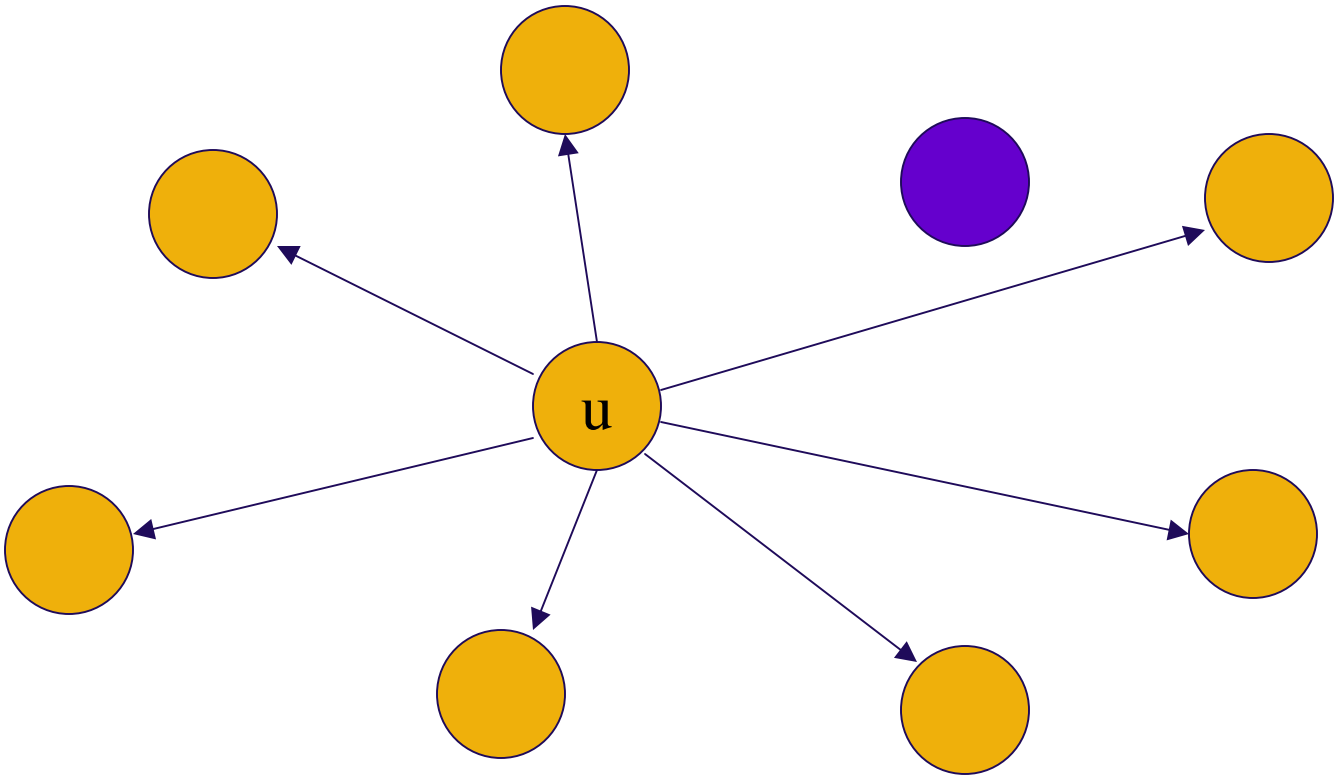
- Esiste un Group Controller (GC)
- Tutti i nodi condividono Traffic Encryption Key (TEK) per cifrare i dati della comunicazione.
- La TEK deve essere aggiornata ogni volta che varia la composizione del gruppo.
- Ogni nodo condivide una Key Encryption Key (KEK) con GC per la cifratura della TEK aggiornata

Traffic Encryption Key

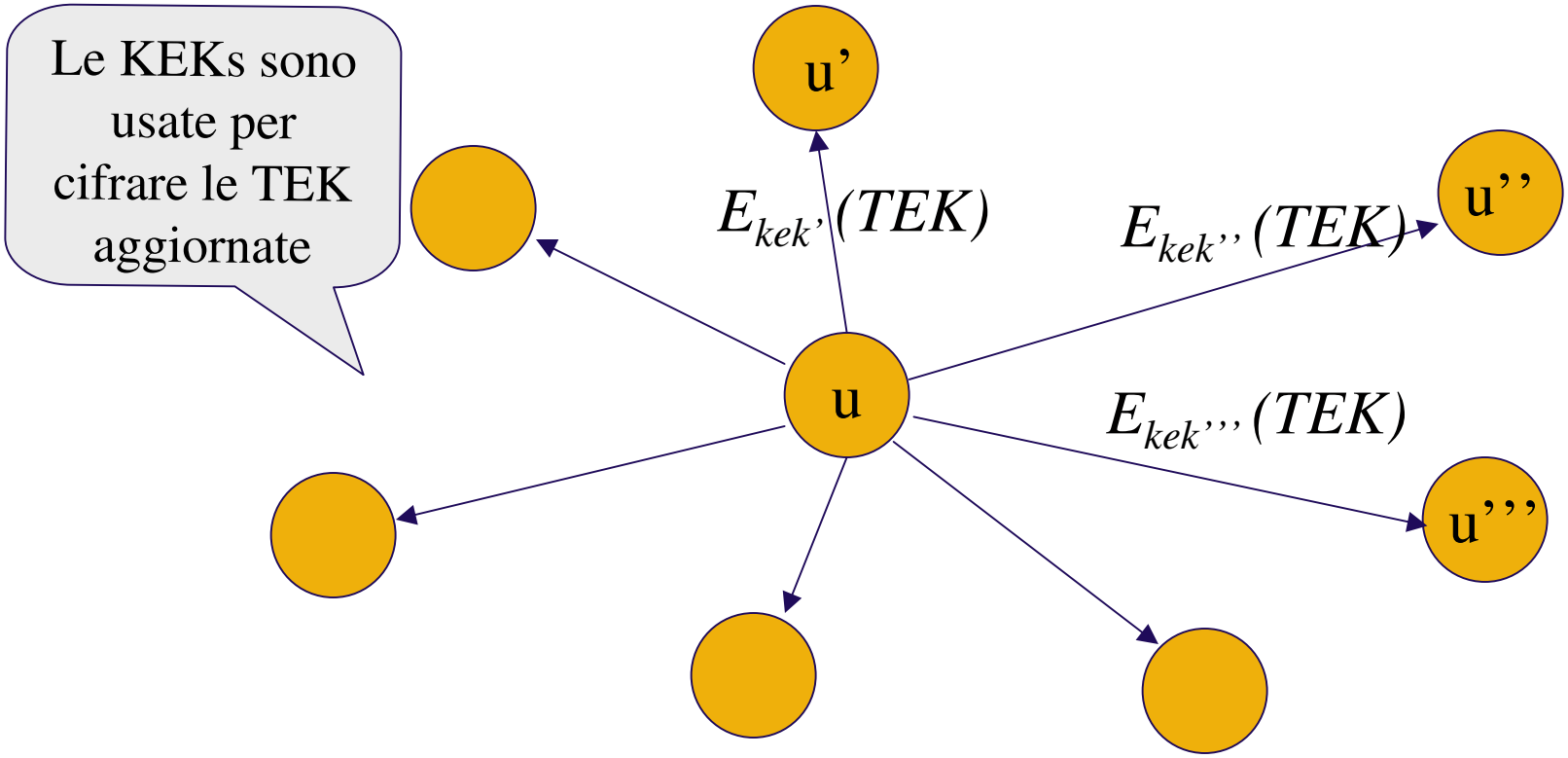


u spedisce un messaggio cifrato con la TEK

Key Encryption Key



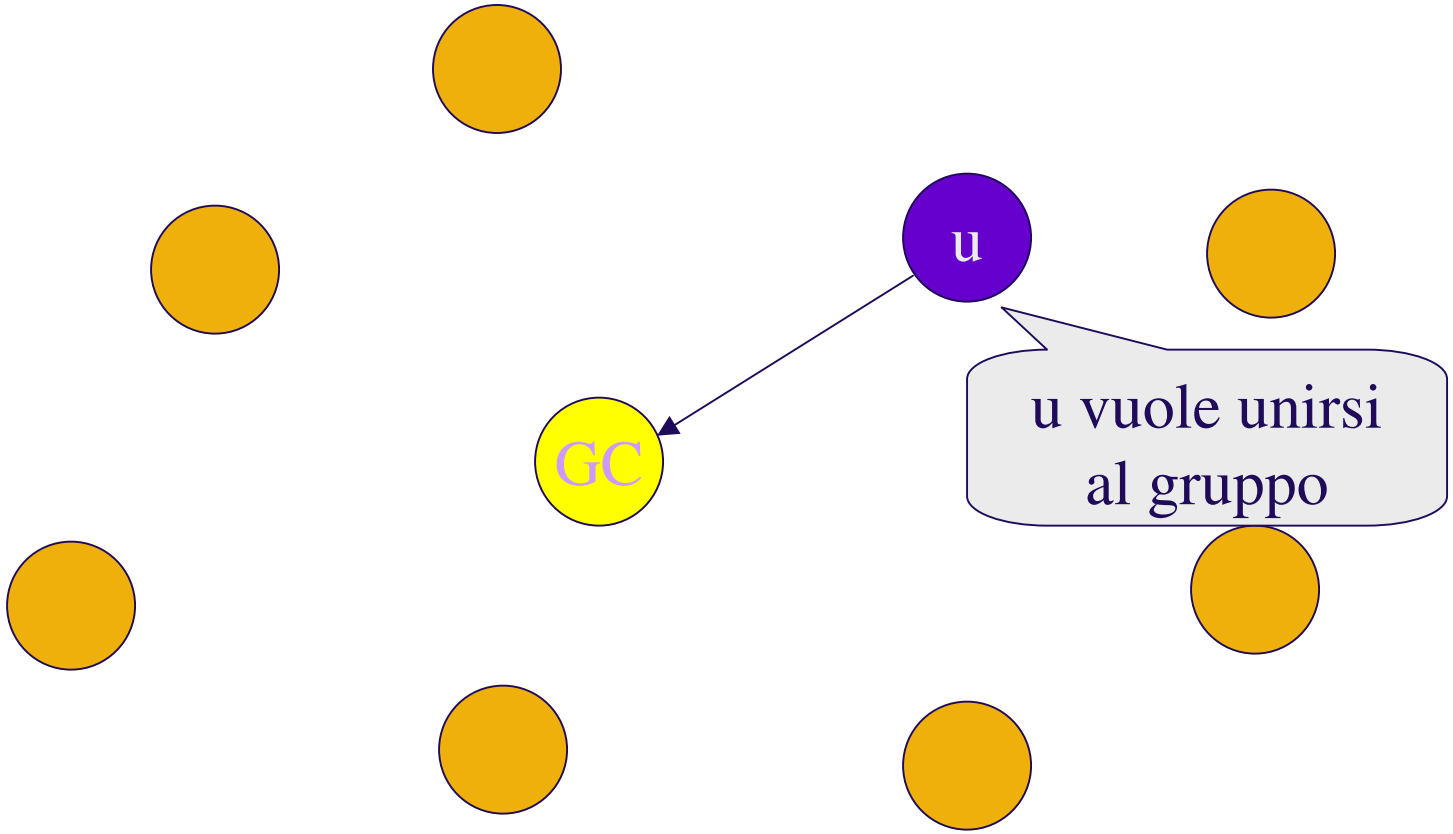
Key Encryption Key



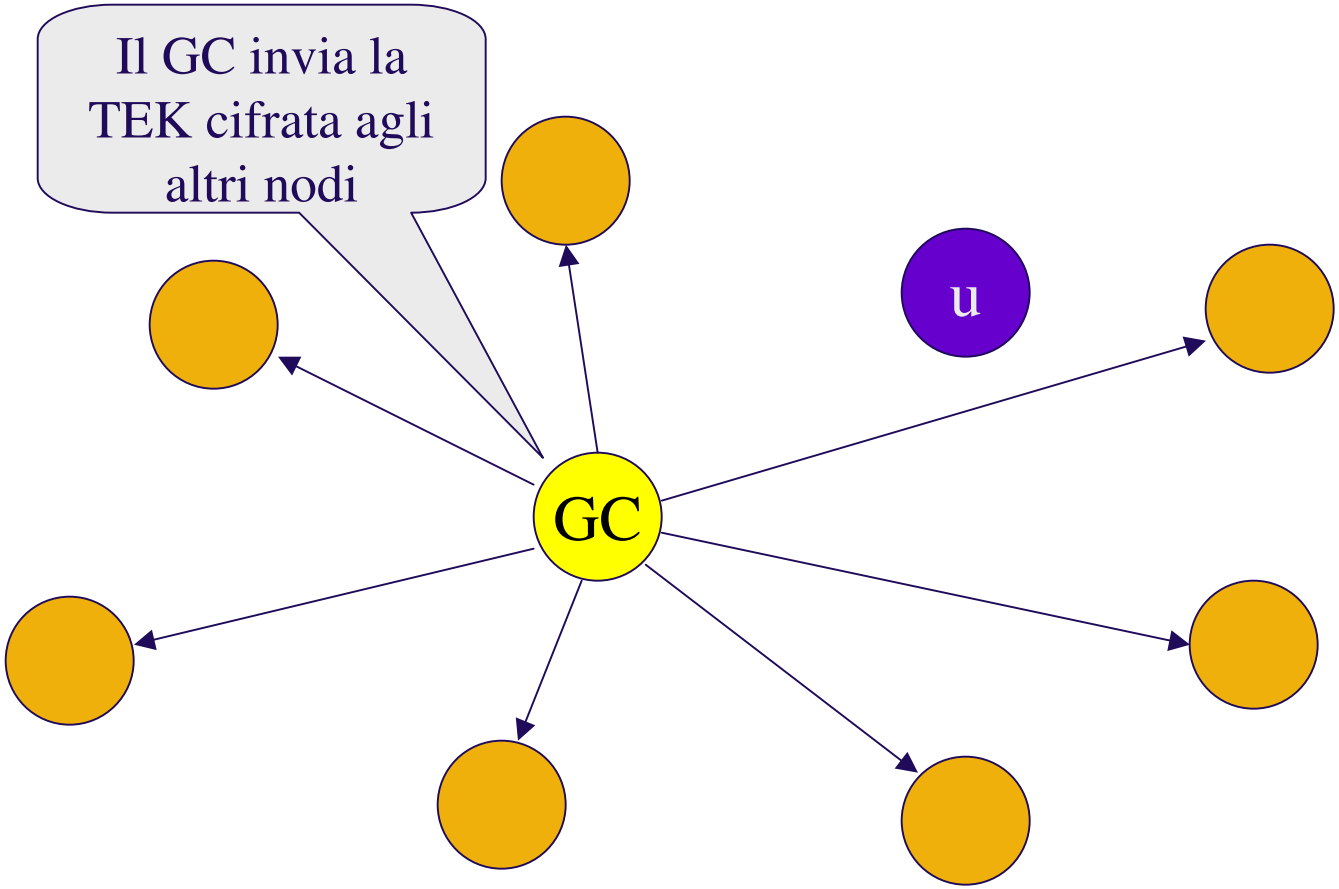
Uno schema semplice per il Re-keying: a Stella

- Ogni utente condivide una KEK segreta con il *GC*
- Quando un utente si aggiunge o lascia il gruppo, il *GC* invia ad ogni nodo un messaggio di re-keying, che viene cifrato con la KEK che ognuno possiede

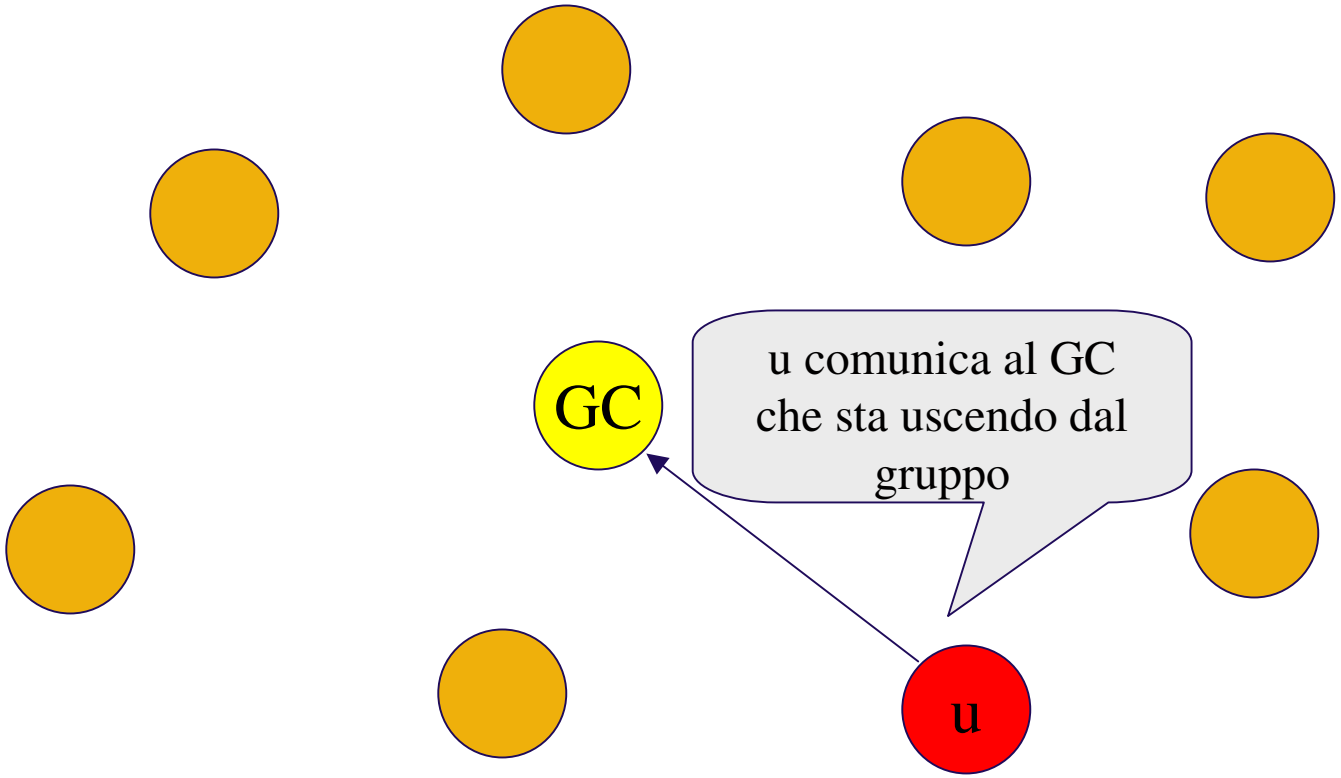
Schema di Re-keying a Stella: Join



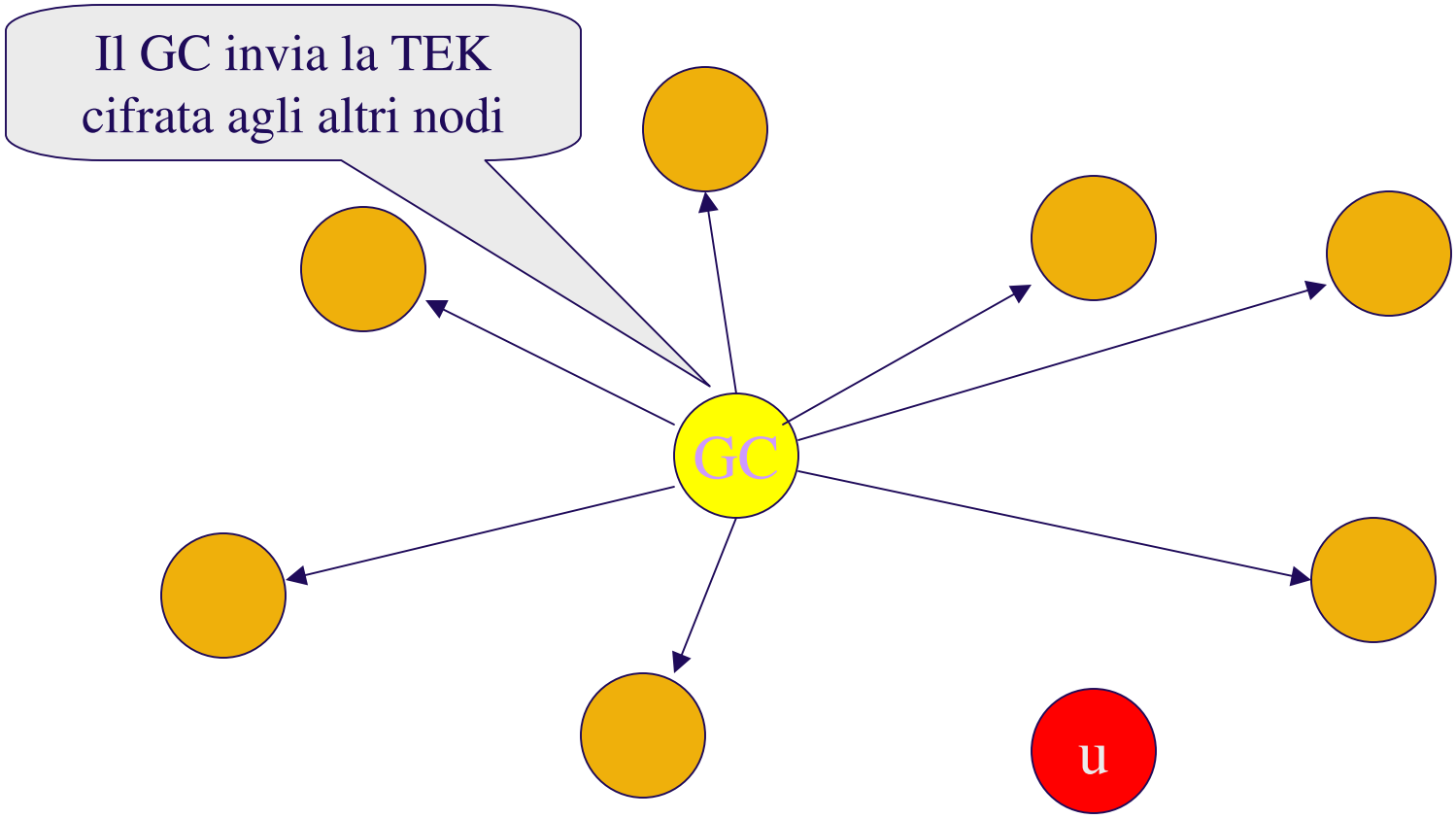
Schema a Stella: Join (Cont.)



Schema a Stella: Leave



Schema a Stella: Leave (Cont.)



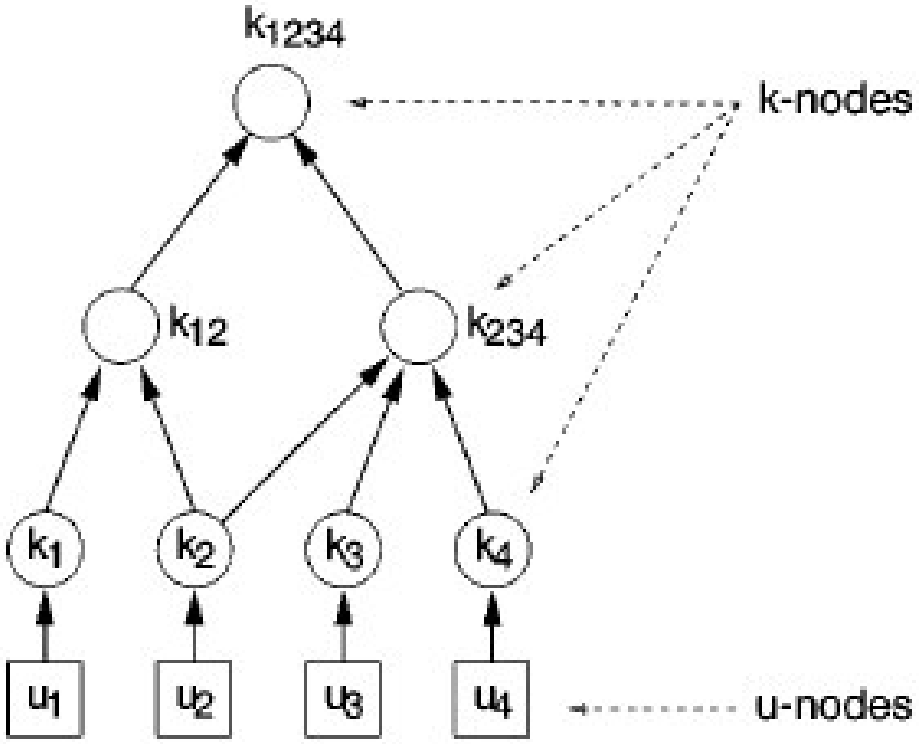
Analisi dello schema a Stella

- Pros:
  - Di facile implementazione
  - Garantisce sia la forward che la backward confidentiality
- Cons:
  - Non è molto scalabile ~ Θ(n) *Oooooops!*

Logical Key Hierarchy (LKH)

- Schema proposto da C.K.Wong, M.Gouda, e S.S.Lam
- Garantisce sia la forward che la backward confidentiality
- E' altamente scalabile ~ $\Theta(\log n)$

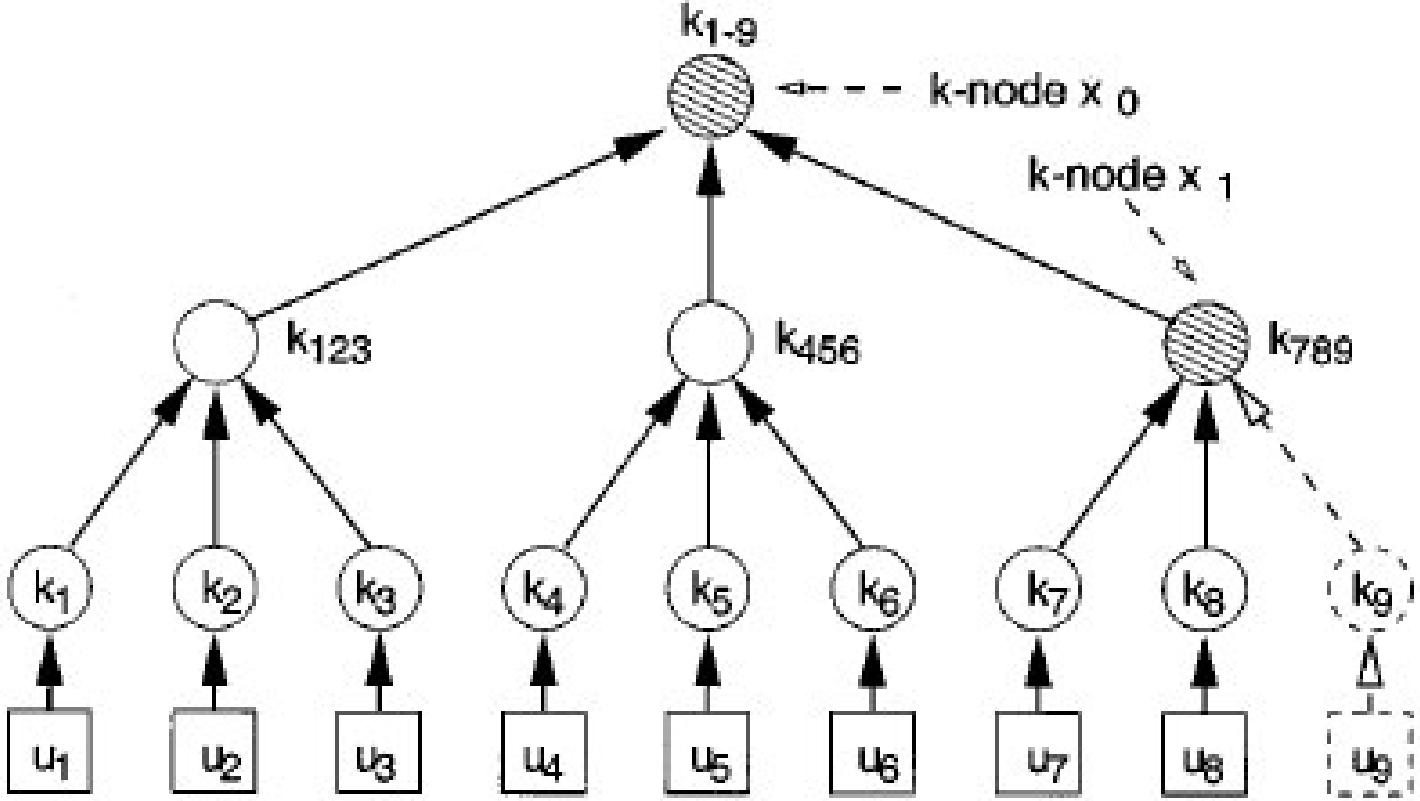
LKH: Grafi delle chiavi



- I nodi *u* sono gli utenti reali
- I nodi *k* rappresentano le chiavi
- *u* conosce *k* se esiste un percorso da *u* a *k*

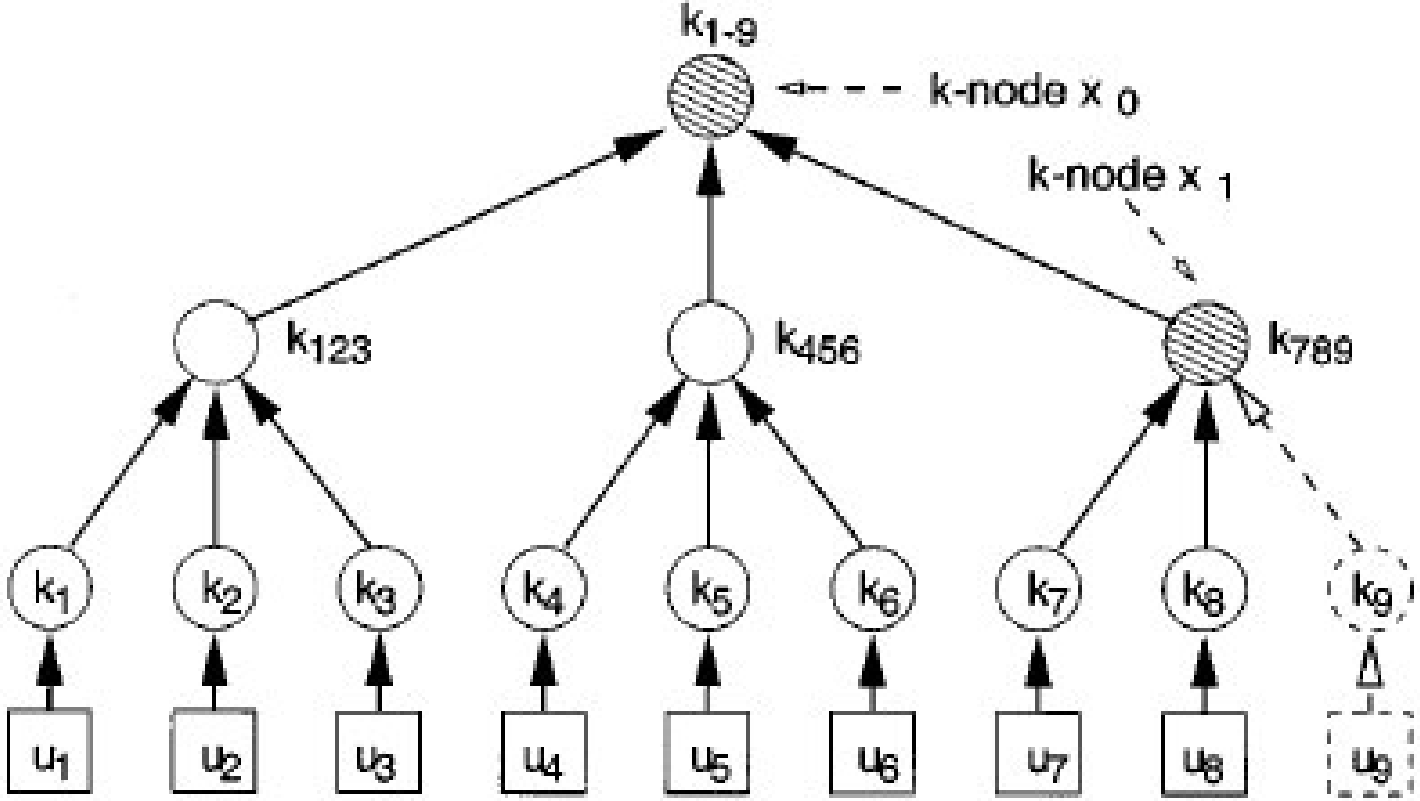
LKH: Join

u_9 vuole aggiungersi al gruppo



LKH: Leave

u_9 vuole lasciare il gruppo



Analisi dello schema LKH

- I messaggi di Re-keying sono inviati bottom-up
- La complessità dipende dall'altezza dell'albero, $\Theta(\log n)$
- Si possono fare delle scelte per il re-keying :
  - user-oriented
  - key-oriented
  - group-oriented

User, Key, or Group?

- Il re-keying User-oriented non è nient'altro che il ragguppamento dei messaggi di re-keying in base agli utenti ~ meno messaggi, ma più lunghi
- Il re-keying Key-oriented consiste nel raggrupparli in base alle chiavi ~ più messaggi, ma più corti
- Il re-keying Group-oriented consiste nel raggruppare tutti i messaggi di re-keying e generarne uno molto grande ~ un unico messaggio gigantesco

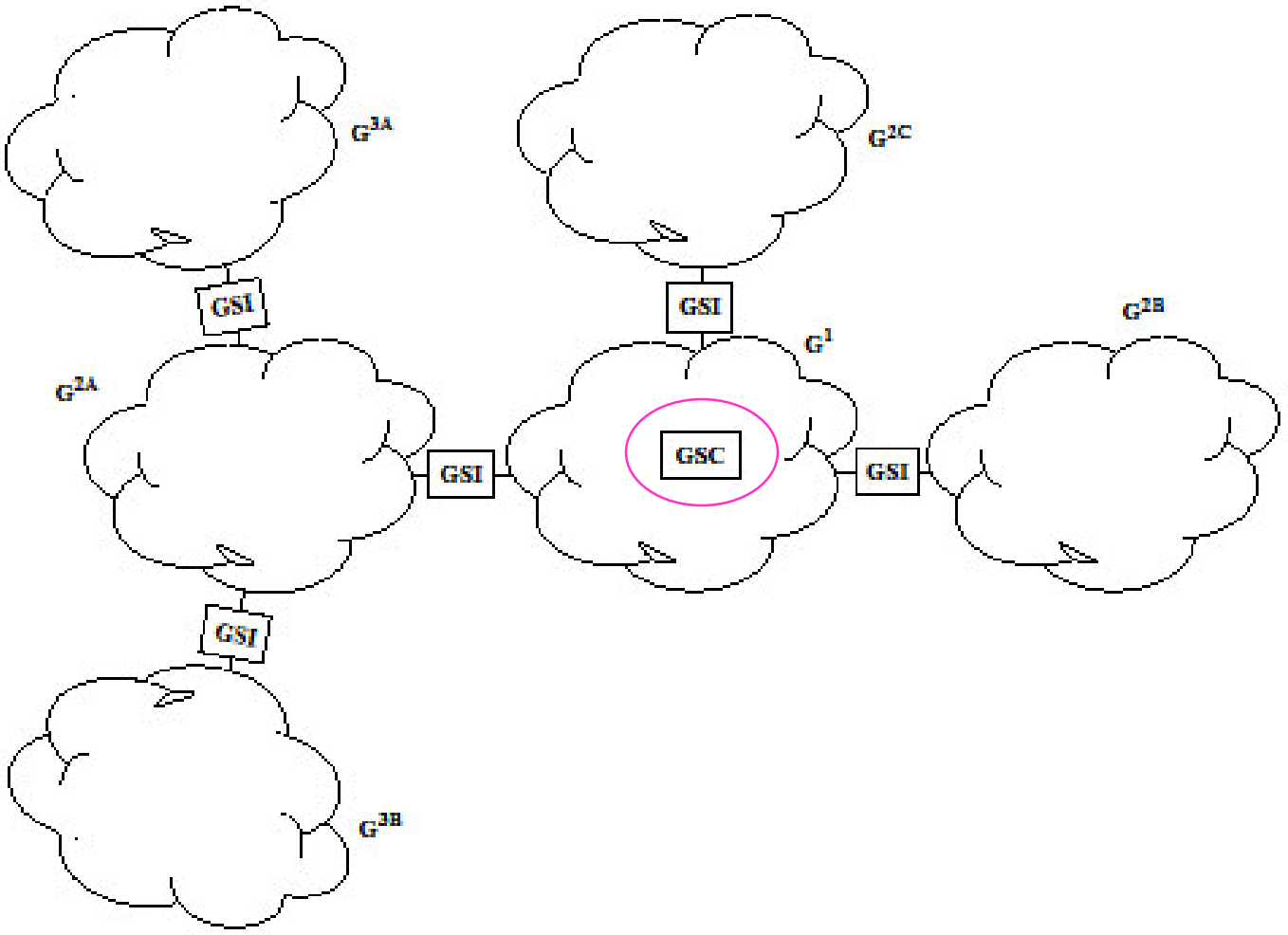
Multicast Sicuro Non-flat

- Il Dominio è suddiviso fisicamente in layers diversi
- Ciascun sotto-gruppo possiede una chiave di encryption per cifrare i propri dati
- E' in genere necessario ritrasmettere i dati
- Esempi: Intra-domain Group Key Management Protocol (IGKMP), Iolus, Kronos

Lo schema Iolus

- Eliminazione completa di un singolo gruppo sicuro di tipo flat
- Si basa su un *albero per la distribuzione sicura*
- Group Security Agent (GSA)
  - Group Security Controller (GSC)
  - Group Security Intermediaries (GSI)
- I GSAs formano una gerarchia di sotto-gruppi

Albero per la Distribuzione Sicura



Albero per la Distribuzione Sicura (Cont.)

- Il *GSC* mantiene il controllo del sotto-gruppo di più alto livello
- I *GSIs* sono servers speciali trusted che sono autorizzati a comportarsi come proxies per il *GSC* o per i nodi *GSIs* di livello più alto
- I *GSIs* formano un bridge tra il nodo di livello più alto ed i sotto-gruppi sottostanti

Albero per la Distribuzione Sicura (Cont.)

- All'interno di ogni sotto-gruppo, c'è un vero e proprio gruppo con chiave k_{GRP} (detta anche k_{SGRP})
- All'interno di ogni sotto-gruppo, il GSA condivide una chiave privata $k_{GSA-MBR}$ con ogni membro del gruppo
- Con riferimento ad ogni sotto-gruppo, tutti i membri formano un blocco per il multicast sicuro di tipo flat, e quindi possono essere usati tutti gli schemi di re-keying come ad esempio l'LKH, LKH+,…

Sommario

- Necessità di gerarchia per assicurare la scalabilità;
- Distinzione tra chiavi usate per il traffico e chiavi usate per la gestione
- Necessità di aggiornare il proprio stato interno