# Estimating the Upper Bound of the Cover Size of the Complete Subtree Method

Big Data Security

李卓倩, 王帅朝, Raziur Rahman Totha, Julien Schmidt

June 2, 2017

Shanghai Jiao Tong University

# The Problem

- Broadcast Encryption
- Stateless Receivers
- Revocation Scheme

**Goal**

Sender sending message to a group of users, in which a subset is considered revoked and should not be able to obtain the content.

# The Problem

- Broadcast Encryption
- Stateless Receivers
- Revocation Scheme

**Goal**

Sender sending message to a group of users, in which a subset is considered revoked and should not be able to obtain the content.

**Example**

Pay-TV

# The Subset-Cover Revocation Framework

D. Naor, M. Naor, and J. Lotspiech. "Revocation and Tracing Schemes for Stateless Receivers". *Annual International Cryptology Conference*. Springer Berlin Heidelberg, 2001.

- Provide condition for the security of revocation schemes
- Separation between long-lived keys and short-lived keys
- Separation of tracing mechanism from revocation algorithm
- Subset-Cover Algorithms

# The Subset-Cover Revocation Framework

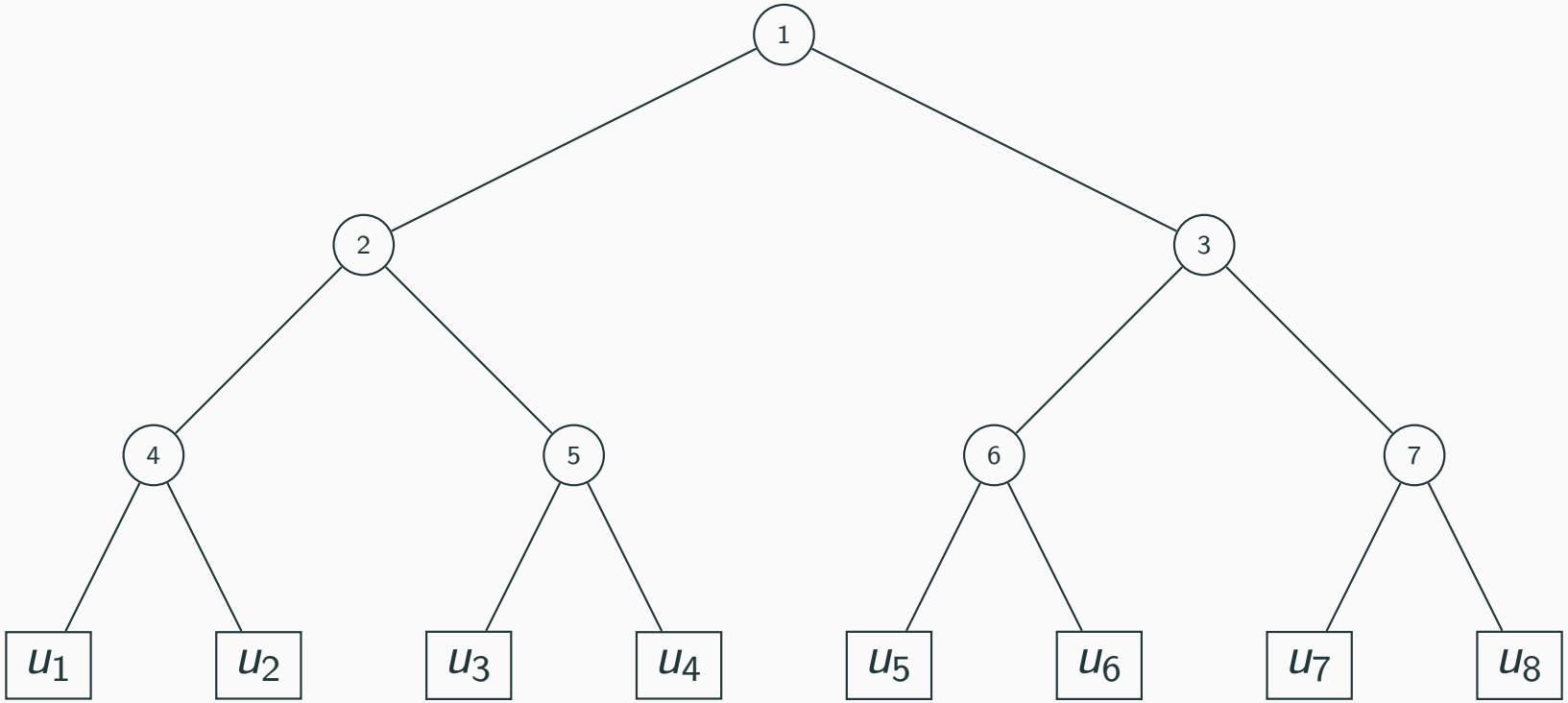
D. Naor, M. Naor, and J. Lotspiech. "Revocation and Tracing Schemes for Stateless Receivers". *Annual International Cryptology Conference*. Springer Berlin Heidelberg, 2001.

- Provide condition for the security of revocation schemes
- Separation between long-lived keys and short-lived keys
- Separation of tracing mechanism from revocation algorithm
- Subset-Cover Algorithms
  1. Receivers partitioned into multiple subsets
  2. Cover all non-revoked receivers with disjoint subsets
  - No assumed upper bound of revoked receivers

| Method | Cover Size | Keys per Receiver | Processing Time | Decryptions |
|---|---|---|---|---|
| Complete Subtree | $r \log \frac{n}{r}$ | $\log n$ | $O(\log \log n)$ | 1 |
| Subset Difference | $2r - 1$ | $\frac{1}{2} \log^2 n$ | $O(\log n)$ | 1 |

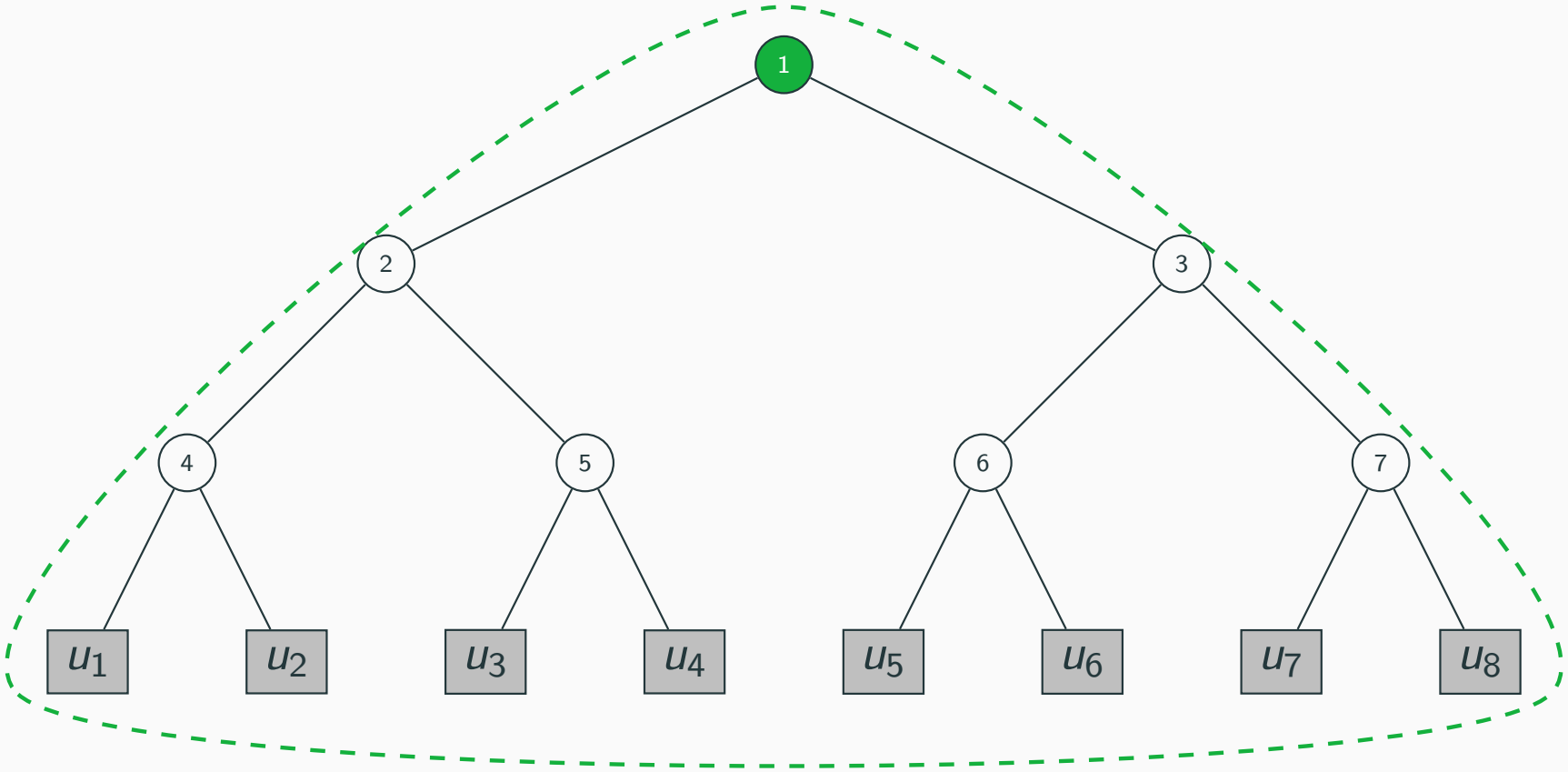
# The Complete Subtree Method

$S_j$ = Set of (sub-) child leaves



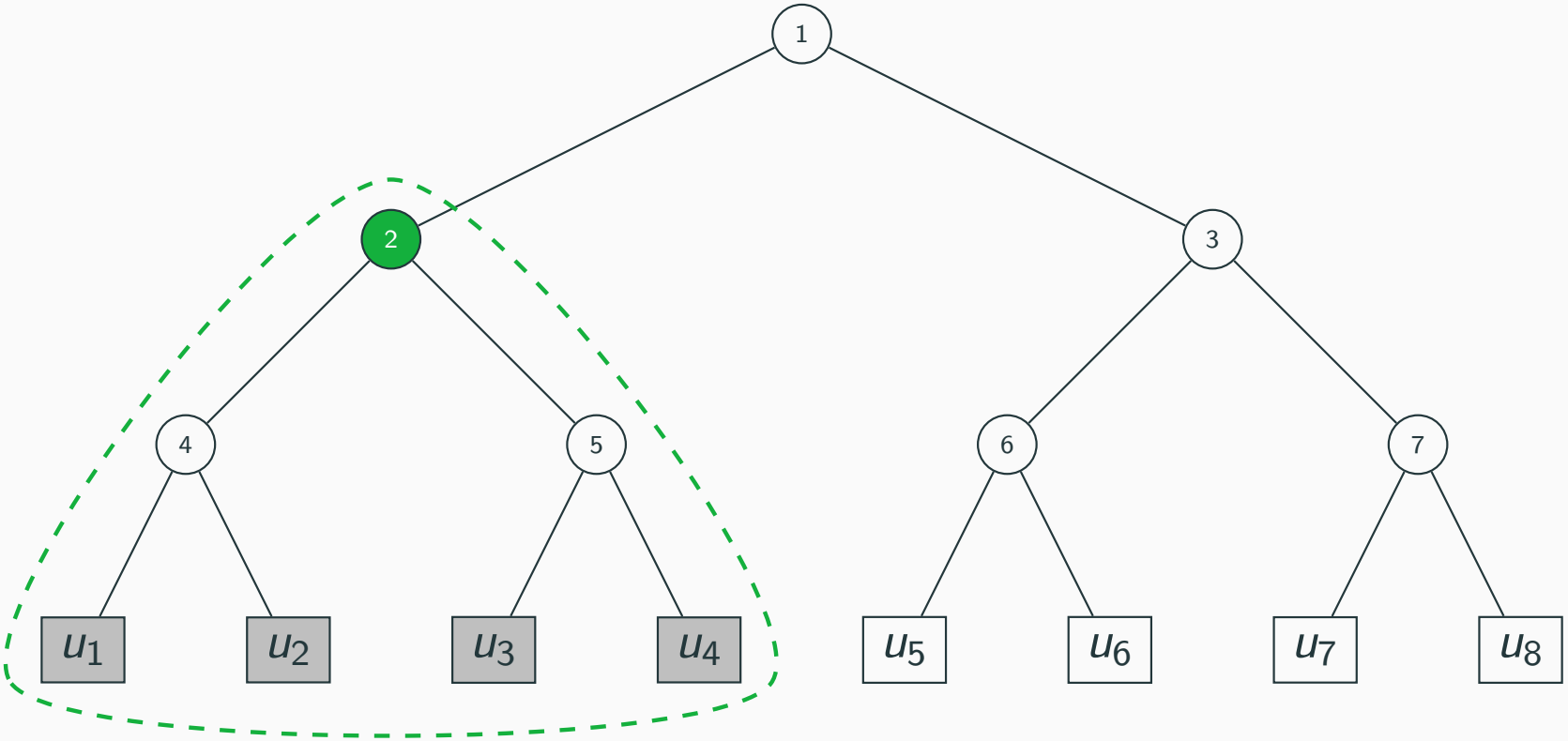
# The Complete Subtree Method

$S_1 = \{u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8\}$



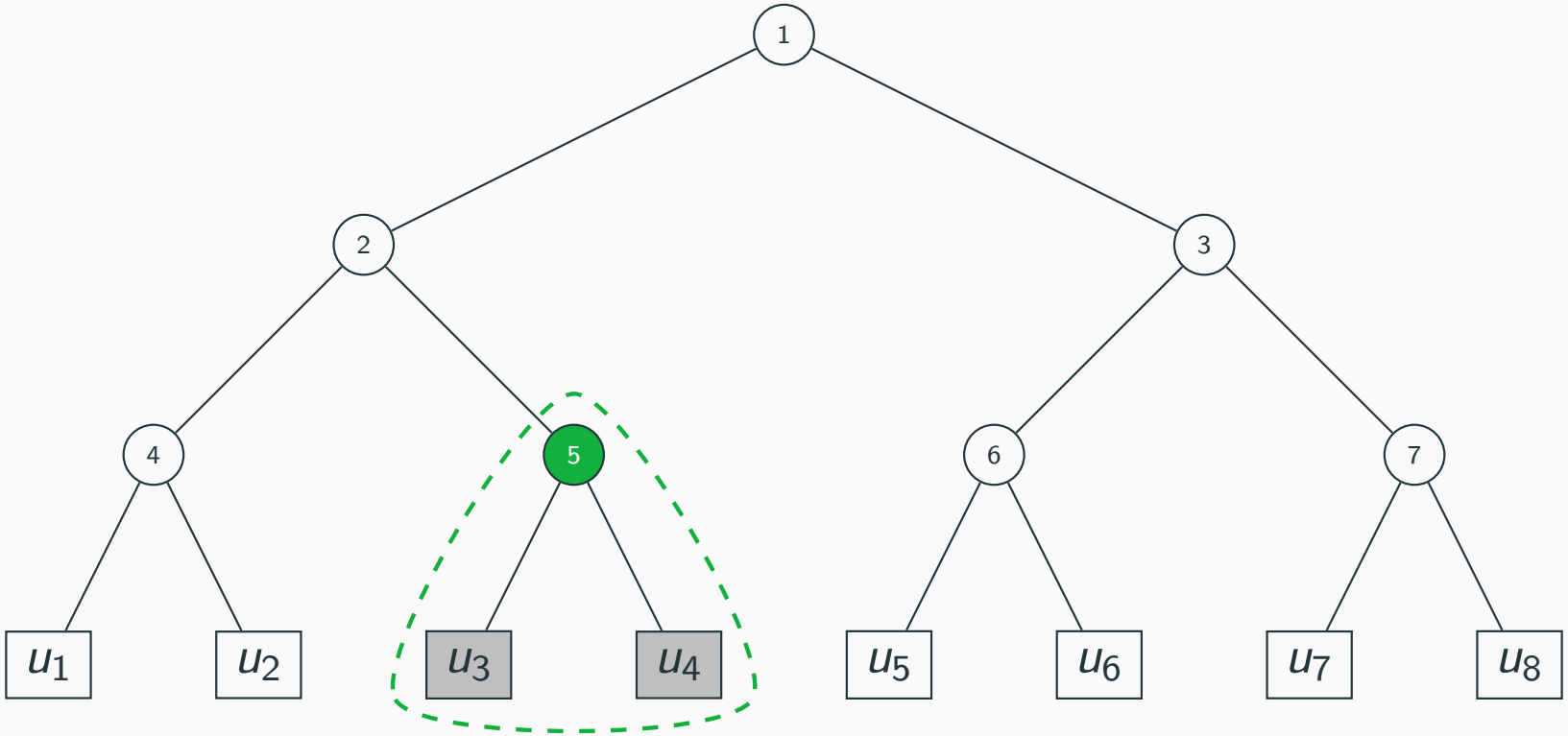
# The Complete Subtree Method

$S_2 = \{u_1, u_2, u_3, u_4\}$



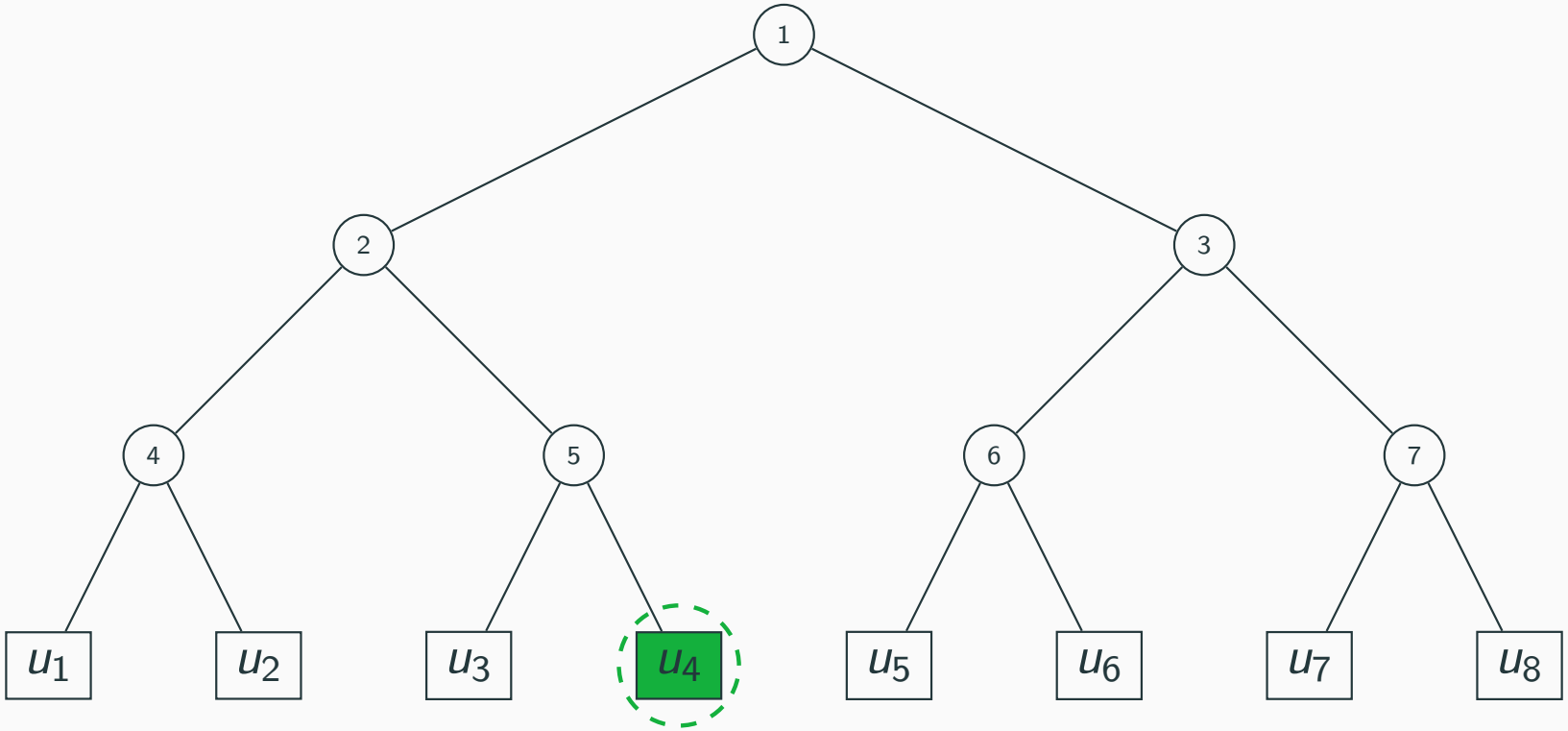
# The Complete Subtree Method

$S_5 = \{u_3, u_4\}$



# The Complete Subtree Method

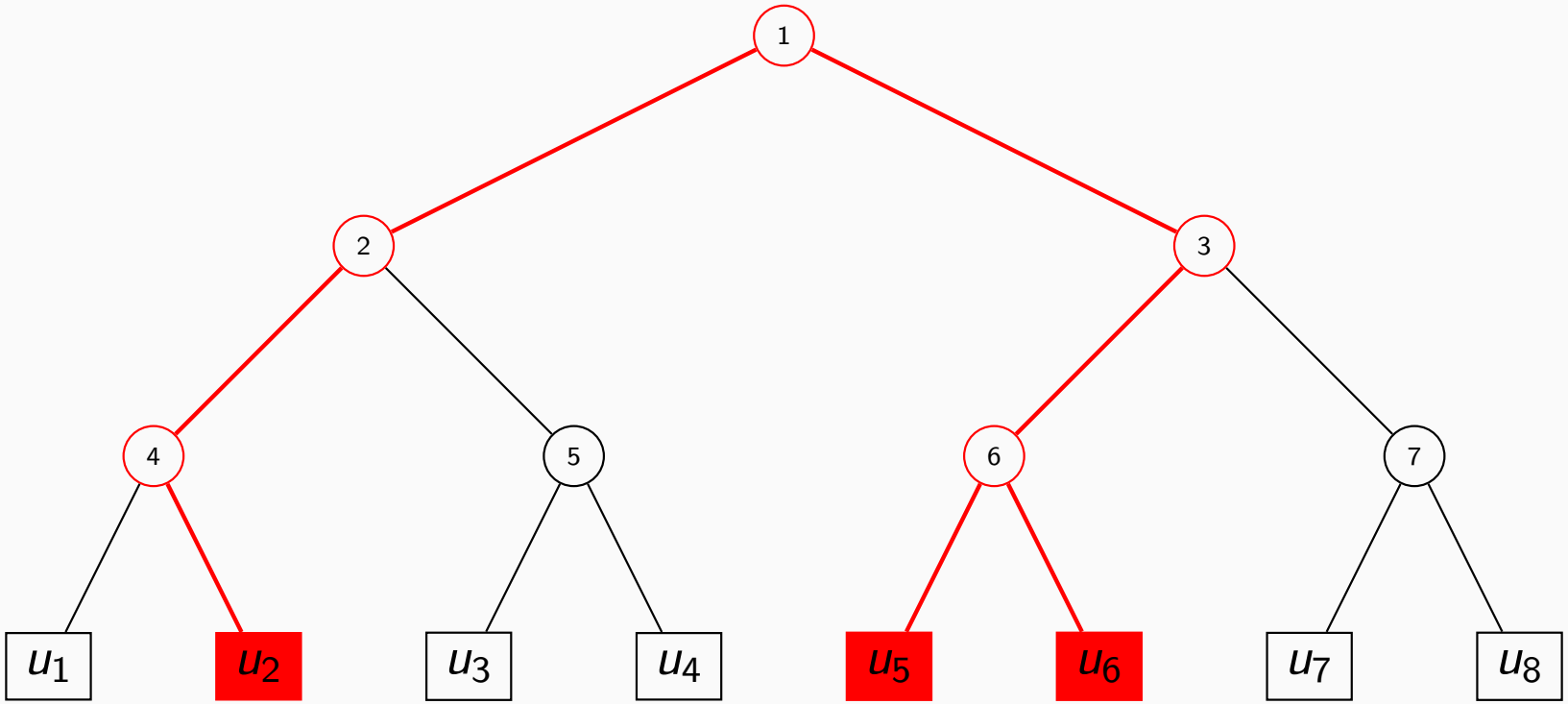
$S_{11} = \{u_4\}$



# The Complete Subtree Method

$\mathcal{R} = \{u_2, u_5, u_6\}$

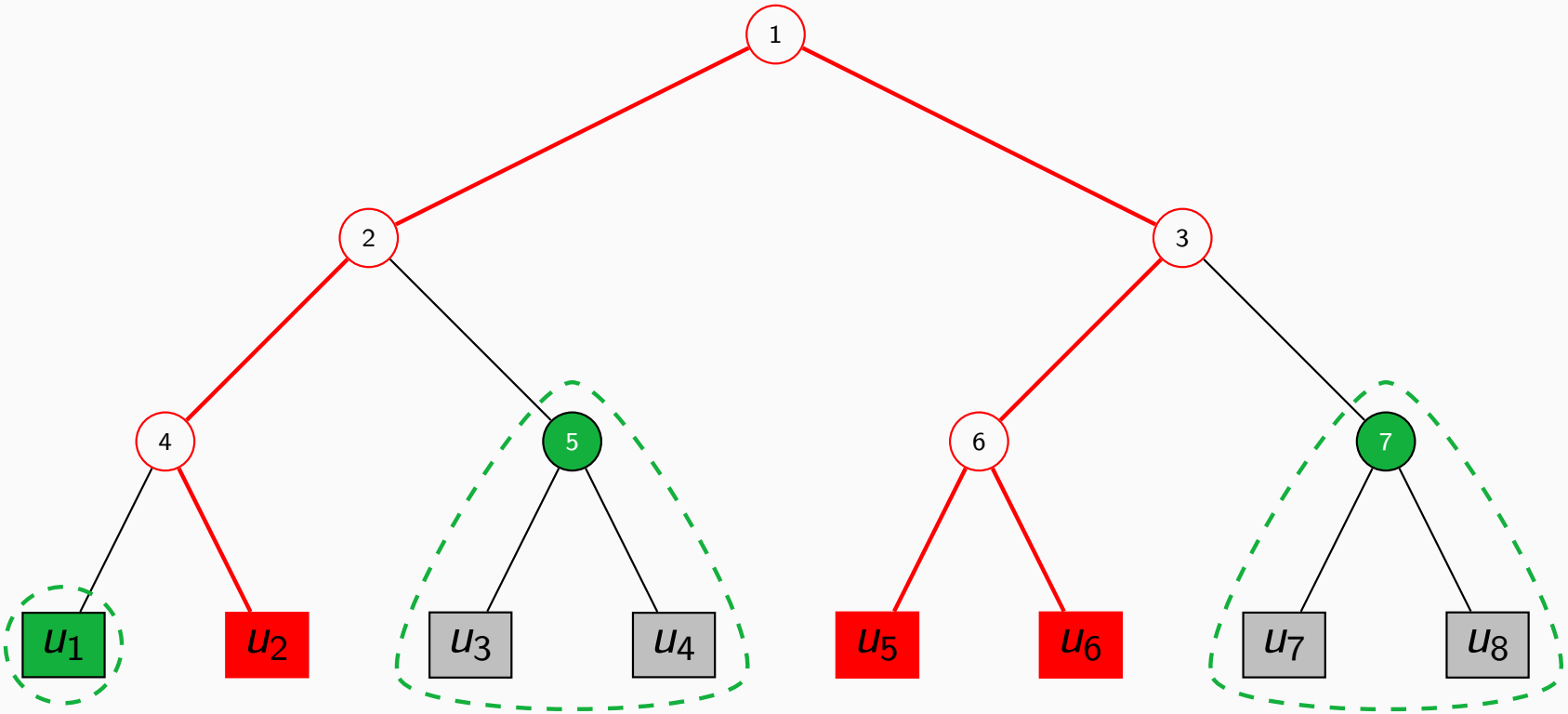
*Steiner Tree* $ST(\mathcal{R})$



# The Complete Subtree Method

$\mathcal{R} = \{u_2, u_5, u_6\}$

$\mathcal{N} \setminus \mathcal{R} = S_8 \cup S_5 \cup S_7$



# The Cover Size Upper Bound

*The cover size is at most*

$$\mathbf{r} \cdot \log \frac{\mathbf{n}}{\mathbf{r}}$$

# The Cover Size Upper Bound

*The cover size is at most*

$$\mathbf{r} \cdot \log \frac{\mathbf{n}}{\mathbf{r}}$$

$\Rightarrow$ How many **subset keys** have to be transmitted in the **worst case**?

# The Cover Size Upper Bound

*The cover size is at most*

$$\mathbf{r} \cdot \log \frac{\mathbf{n}}{\mathbf{r}}$$

$\Rightarrow$ How many **subset keys** have to be transmitted in the **worst case**?

| | |
|---|---|
| $\mathcal{N}$ | Set of receivers (devices) |
| $S_1, ..., S_w$, $S_j \subseteq \mathcal{N}$ | Subsets |
| $\mathcal{R}$ | Set of revoked receivers |
| $\mathcal{N} \setminus \mathcal{R} = \bigcup_{j=1}^{m} S_{i_j}$ | Set of non-revoked receivers is a partition of subsets |
| $n = \lvert\mathcal{N}\rvert$ | |
| $r = \lvert\mathcal{R}\rvert$ | |

# Proof Idea

**Observation**

The number of required subsets is equal to the number of nodes with degree 1 in $ST(\mathcal{R})$

**Upper Bound $C_{max}(n, r)$:**

The max. cover size for $n$ receivers in total of which $r$ are revoked

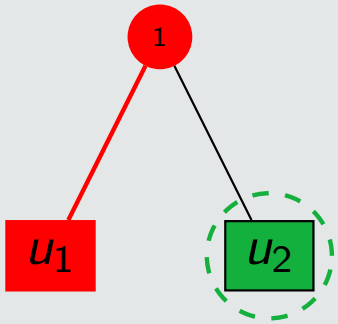
- Trivially $C_{max}(n, 0) = 1$
- $0 < r \leq n \implies$ Induction

# Induction Base

**Depth $k = 1$:**

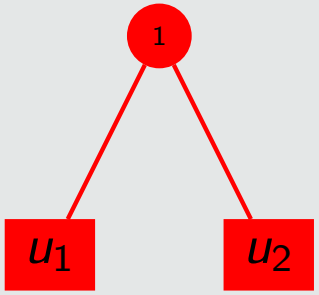
$n = 2^k = 2$

$\mathbf{r = 1}$:




$C_{max}(2, 1) = 1 = 1 \cdot \log \frac{2}{1}$

$\mathbf{r = 2}$:

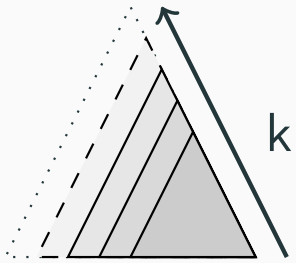



$C_{max}(2, 2) = 0 = 2 \cdot \log \frac{2}{2}$

# Induction Hypothesis

To prove: $r \cdot \log \frac{n}{r}$

- $k = \log(n)$
- $\log(\frac{n}{r}) = \log(n) - \log(r) = k - \log(r)$

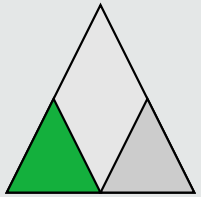



$$C_{max}(2^k, r) \leq r \cdot (k - log(r))$$

# Induction Step

$k \to k+1$:

**Case 1: All in one side**

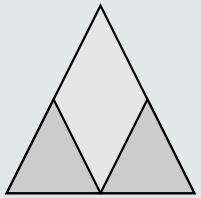


If all leaves are in a subtree of depth $k$, then the total number of nodes of degree 1 is at most:

$$C_{max}(2^{k+1}, r) = r \cdot (k - \log r) + 1 \leq r \cdot (k + 1 - \log r) \tag{1}$$

# Induction Step

**Case 2:** $\mathcal{R} = \mathcal{R}_1 \cup \mathcal{R}_2$



$\mathcal{R}$ is split among the left ($\mathcal{R}_1$) and right subtree ($\mathcal{R}_2$) of the root. Thus:

$$r = r_1 + r_2 \tag{2}$$

By induction at most: $C_{max}(2^{k+1}, r) =$

$$\begin{aligned} r_1 \cdot (k - \log r_1) + r_2 \cdot (k - \log r_2) &= r \cdot k - (r_1 \log r_1 + r_2 \log r_2) \\ &\leq r \cdot (k + 1 - \log r) \end{aligned} \tag{3}$$

since $(r_1 \log r_1 + r_2 \log r_2) \geq r \cdot (\log r - 1)$

■

**Questions?**