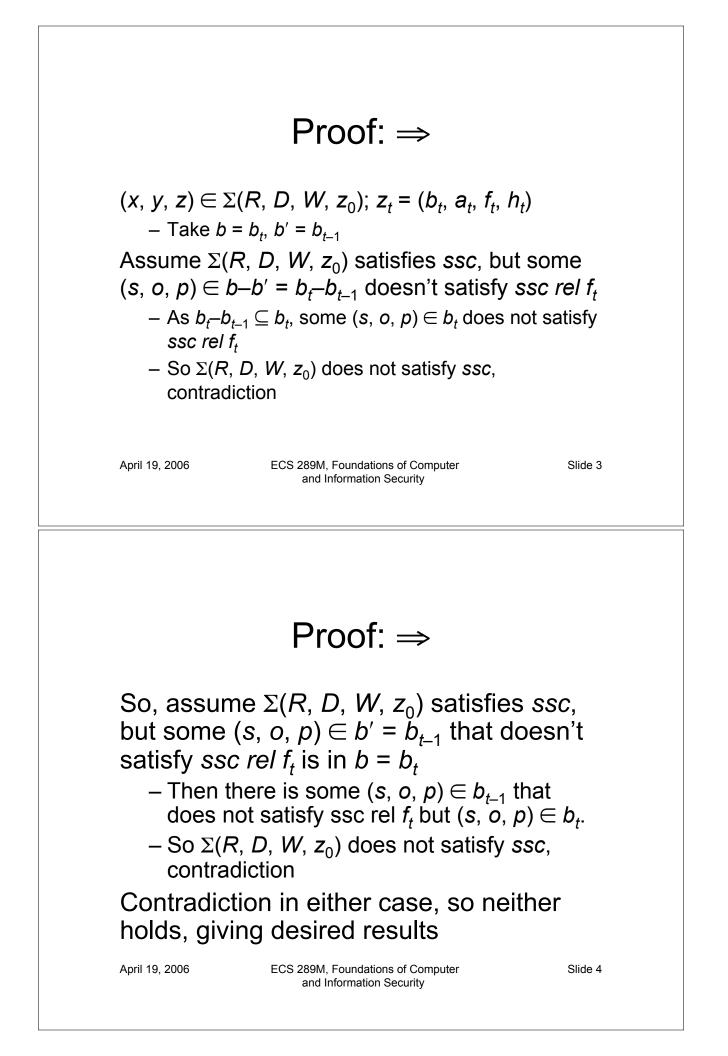
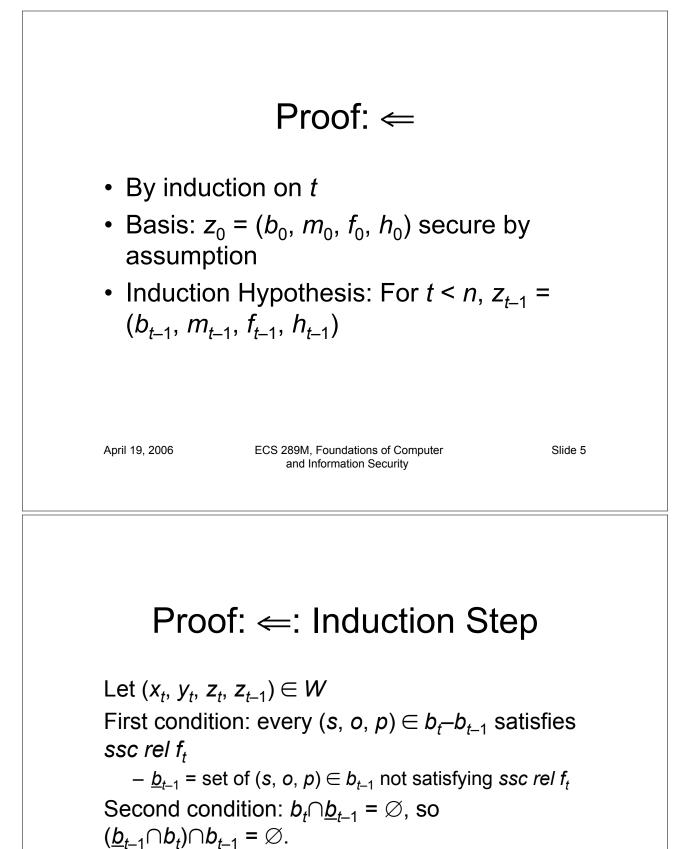
ECS 289M Lecture 9

April 19, 2006

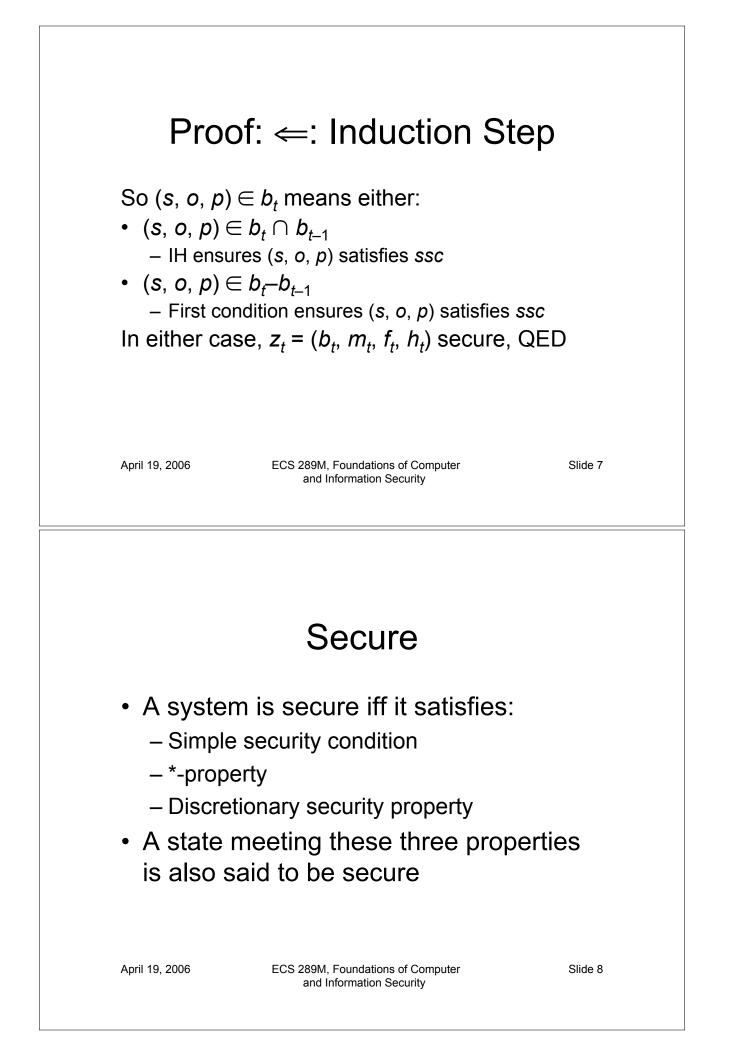
Necessary and Sufficient

- Σ(R, D, W, z₀) satisfies the simple security condition for any secure state z₀ iff for every action (r, d, (b, m, f, h), (b', m', f', h')), W satisfies
 - Every $(s, o, p) \in b b'$ satisfies ssc rel f
 - Every $(s, o, p) \in b'$ that does not satisfy *ssc rel f* is not in *b*
- Note: "secure" means z₀ satisfies ssc rel f
- First says every (s, o, p) added satisfies ssc rel f; second says any (s, o, p) in b' that does not satisfy ssc rel f is deleted





If $(s, o, p) \in b_t \cap b_{t-1}$, this means $(s, o, p) \notin \underline{b}_{t-1}$, so (s, o, p) satisfies *ssc rel* f_t





 Σ(R, D, W, z₀) is a secure system if z₀ is a secure state and W satisfies the conditions for the preceding three theorems

 The theorems are on the slides titled "Necessary and Sufficient"

April 19, 2006

ECS 289M, Foundations of Computer and Information Security Slide 9

Rule

- $\rho: R \times V \rightarrow D \times V$
- Takes a state and a request, returns a decision and a (possibly new) state
- Rule ρ ssc-preserving if for all (r, v) ∈R × V and v satisfying ssc rel f, ρ(r, v) = (d, v') means that v' satisfies ssc rel f'.
 - Similar definitions for *-property, ds-property
 - If rule meets all 3 conditions, it is security-preserving

Unambiguous Rule Selection

Problem: multiple rules may apply to a request in a state

- if two rules act on a read request in state v ...

- Solution: define relation W(ω) for a set of rules ω = { ρ
 1, ..., ρ_m } such that a state (r, d, v, v') ∈W(ω) iff either
 d = i; or
 - for exactly one integer *j*, $\rho_i(r, v) = (d, v')$

• Either request is illegal, or only one rule applies

April 19, 2006

ECS 289M, Foundations of Computer and Information Security Slide 11

Rules Preserving SSC

- Let ω be set of ssc-preserving rules. Let state z₀ satisfy simple security condition. Then Σ(R, D, W(ω), z₁) satisfies simple security condition
 - z_0) satisfies simple security condition
 - Proof: by contradiction.
 - Choose (x, y, z) ∈ Σ(R, D, W(ω), z₀) as state not satisfying simple security condition; then choose t ∈ N such that (x_t, y_t, z_t) is first appearance not meeting simple security condition
 - As $(x_t, y_t, z_t, z_{t-1}) \in W(\omega)$, there is unique rule $\rho \in \omega$ such that ρ $(x_t, z_{t-1}) = (y_t, z_t)$ and $y_t \neq \underline{i}$.
 - As ρ ssc-preserving, and z_{t-1} satisfies simple security condition, then z_t meets simple security condition, contradiction.

Adding States Preserving SSC

- Let v = (b, m, f, h) satisfy simple security condition. Let (s, o, p)
 ∉ b, b' = b ∪ { (s, o, p) }, and v' = (b', m, f, h). Then v' satisfies simple security condition iff:
 - 1. Either $p = \underline{e}$ or $p = \underline{a}$; or
 - 2. Either $p = \underline{r}$ or $p = \underline{w}$, and $f_c(s) \text{ dom } f_o(o)$

– Proof

- Immediate from definition of simple security condition and v' satisfying ssc rel f
- 2. v' satisfies simple security condition means $f_c(s)$ dom $f_o(o)$, and for converse, $(s, o, p) \in b'$ satisfies ssc rel f, so v' satisfies simple security condition

April 19, 2006

ECS 289M, Foundations of Computer and Information Security Slide 13

Rules, States Preserving *-Property

- Let ω be set of *-property-preserving rules, state z₀ satisfies *-property. Then Σ(R, D, W(ω), z₀) satisfies *-property
- Let v = (b, m, f, h) satisfy *-property. Let (s, o, p) ∉
 b, b' = b ∪ { (s, o, p) }, and v' = (b', m, f, h). Then v' satisfies *-property iff one of the following holds:
 - 1. *p* = <u>e</u>
 - 2. $p = \underline{r}$ and $f_c(s)$ dom $f_o(o)$
 - 3. $p = \underline{w}$ and $f_c(s) = f_o(o)$
 - 4. $p = \underline{a}$ and $f_o(o)$ dom $f_c(s)$

April 19, 2006

ECS 289M, Foundations of Computer and Information Security Slide 14

Rules, States Preserving ds-Property

- Let ω be set of ds-property-preserving rules, state z₀ satisfies ds-property. Then Σ(R, D, W(ω), z₀) satisfies ds-property
- Let v = (b, m, f, h) satisfy ds-property. Let (s, o, p) ∉ b, b' = b ∪ { (s, o, p) }, and v' = (b', m, f, h). Then v' satisfies ds-property iff p ∈ m[s, o].

April 19, 2006

ECS 289M, Foundations of Computer and Information Security Slide 15

Combining

- Let ρ be a rule and $\rho(r, v) = (d, v')$, where v = (b, m, f, h) and v' = (b', m', f', h'). Then:
 - 1. If $b' \subseteq b$, f' = f, and v satisfies the simple security condition, then v' satisfies the simple security condition
 - 2. If $b' \subseteq b$, f' = f, and v satisfies the *-property, then v' satisfies the *-property
 - 3. If $b' \subseteq b$, $m[s, o] \subseteq m'[s, o]$ for all $s \in S$ and $o \in O$, and v satisfies the ds-property, then v' satisfies the ds-property

Proof

1. Suppose *v* satisfies simple security property.

- a) $b' \subseteq b$ and $(s, o, \underline{r}) \in b'$ implies $(s, o, \underline{r}) \in b$
- b) $b' \subseteq b$ and $(s, o, \underline{w}) \in b'$ implies $(s, o, \underline{w}) \in b$
- c) So $f_c(s)$ dom $f_o(o)$
- d) But f' = f
- e) Hence $f'_c(s) \operatorname{dom} f'_o(o)$
- f) So v' satisfies simple security condition

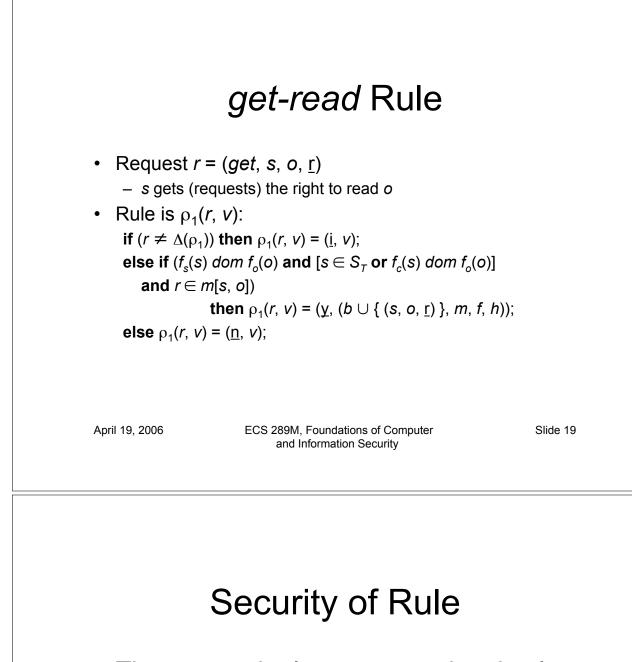
2, 3 proved similarly

April 19, 2006

ECS 289M, Foundations of Computer and Information Security Slide 17

Example Instantiation: Multics

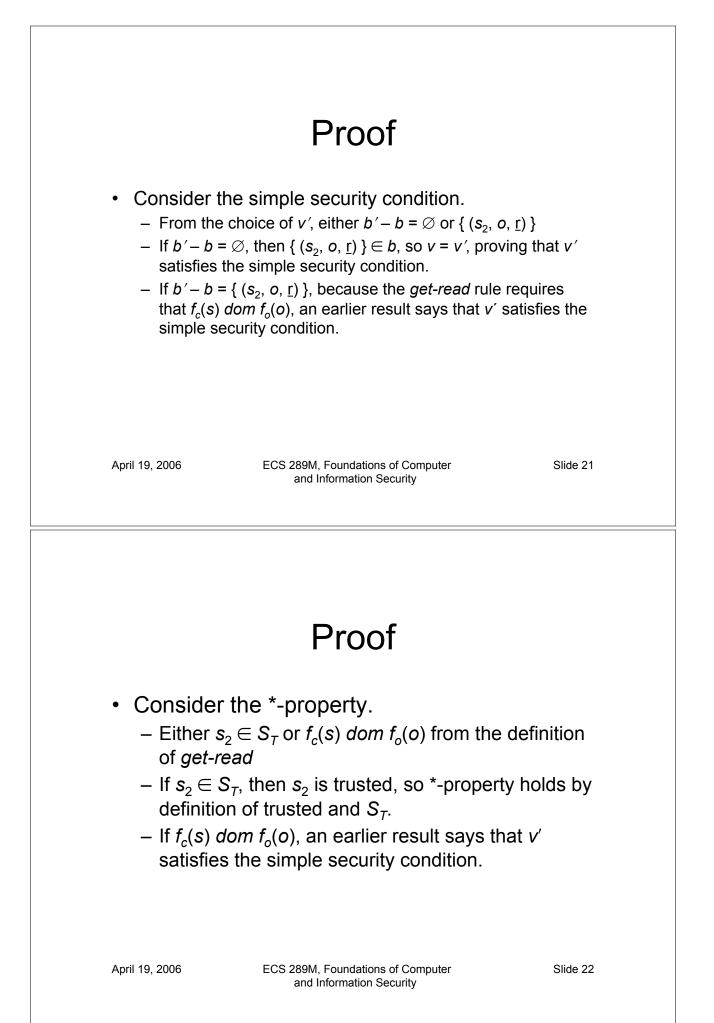
- 11 rules affect rights:
 - set to request, release access
 - set to give, remove access to different subject
 - set to create, reclassify objects
 - set to remove objects
 - set to change subject security level
- Set of "trusted" subjects $S_T \subseteq S$
 - *-property not enforced; subjects trusted not to violate
- $\Delta(\rho)$ domain
 - determines if components of request are valid



 The get-read rule preserves the simple security condition, the *-property, and the ds-property

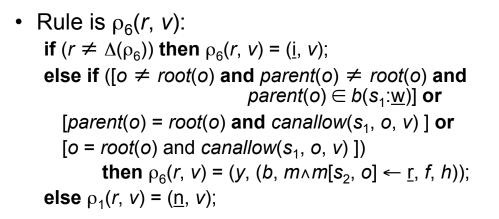
– Proof

• Let *v* satisfy all conditions. Let $\rho_1(r, v) = (d, v')$. If v' = v, result is trivial. So let $v' = (b \cup \{ (s_2, o, \underline{r}) \}, m, f, h)$.



Proof Consider the discretionary security property. - Conditions in the *get-read* rule require $\underline{r} \in m[s, o]$ and either $b' - b = \emptyset$ or { (s_2, o, \underline{r}) } - If $b' - b = \emptyset$, then { (s_2, o, \underline{r}) } $\in b$, so v = v', proving that v'satisfies the simple security condition. - If $b' - b = \{ (s_2, o, \underline{r}) \}$, then $\{ (s_2, o, \underline{r}) \} \notin b$, an earlier result says that v' satisfies the ds-property. April 19, 2006 ECS 289M. Foundations of Computer Slide 23 and Information Security give-read Rule • Request $r = (s_1, give, s_2, o, \underline{r})$ $-s_1$ gives (request to give) s_2 the (discretionary) right to read o Rule: can be done if giver can alter parent of object · If object or parent is root of hierarchy, special authorization required Useful definitions ٠ - root(o): root object of hierarchy h containing o - parent(o): parent of o in h (so $o \in h(parent(o))$) canallow(s, o, v): s specially authorized to grant access when object or parent of object is root of hierarchy $- m \wedge m[s, o] \leftarrow \underline{r}$: access control matrix m with \underline{r} added to m[s, o]April 19, 2006 ECS 289M, Foundations of Computer Slide 24 and Information Security

give-read Rule

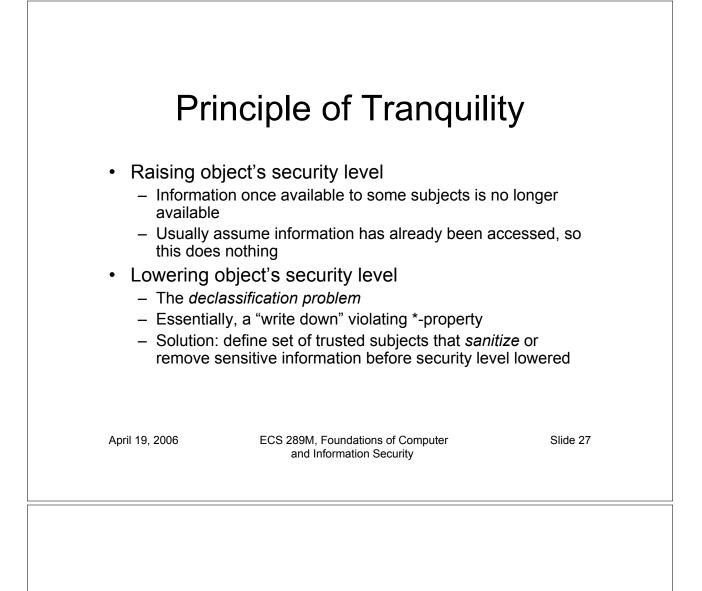


April 19, 2006

ECS 289M, Foundations of Computer and Information Security Slide 25

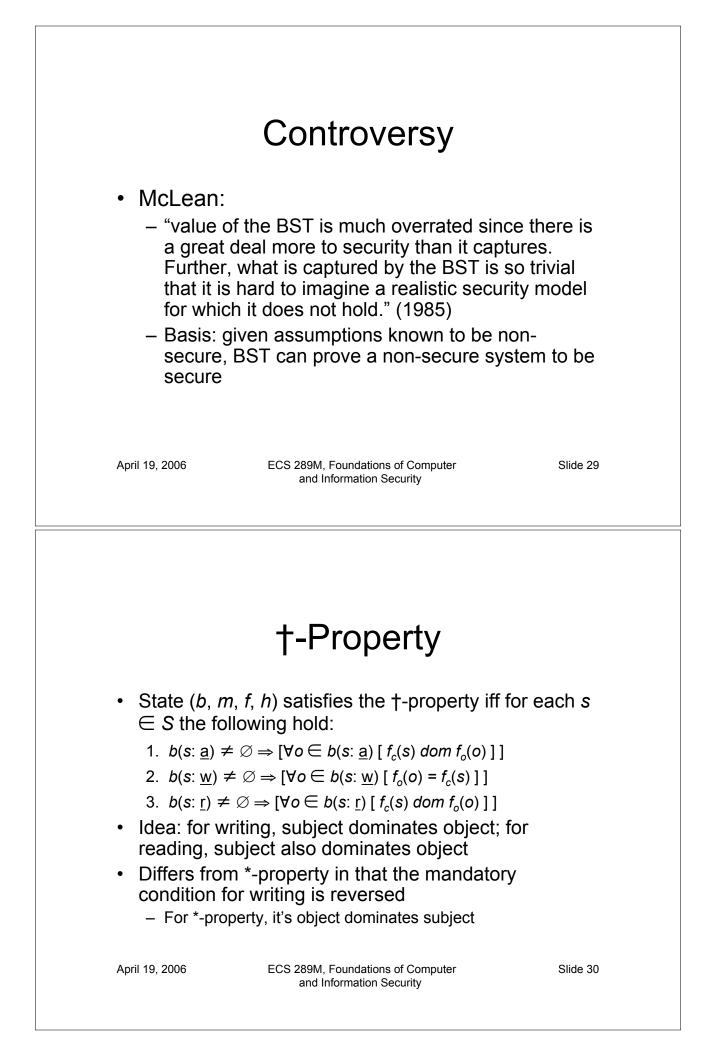
Security of Rule

- The *give-read* rule preserves the simple security condition, the *-property, and the ds-property
 - Proof: Let *v* satisfy all conditions. Let $\rho_1(r, v) = (d, v')$. If v' = v, result is trivial. So let $v' = (b, m[s_2, o] \leftarrow \underline{r}, f, h)$. So b' = b, f' = f, m[x, y] = m'[x, y] for all $x \in S$ and $y \in O$ such that $x \neq s$ and $y \neq o$, and $m[s, o] \subseteq m[s, o]$. Then by earlier result, v' satisfies the simple security condition, the *-property, and the ds-property.



Types of Tranquility

- Strong Tranquility
 - The clearances of subjects, and the classifications of objects, do not change during the lifetime of the system
- Weak Tranquility
 - The clearances of subjects, and the classifications of objects, do not change in a way that violates the simple security condition or the *-property during the lifetime of the system



Analogues

The following two theorems can be proved

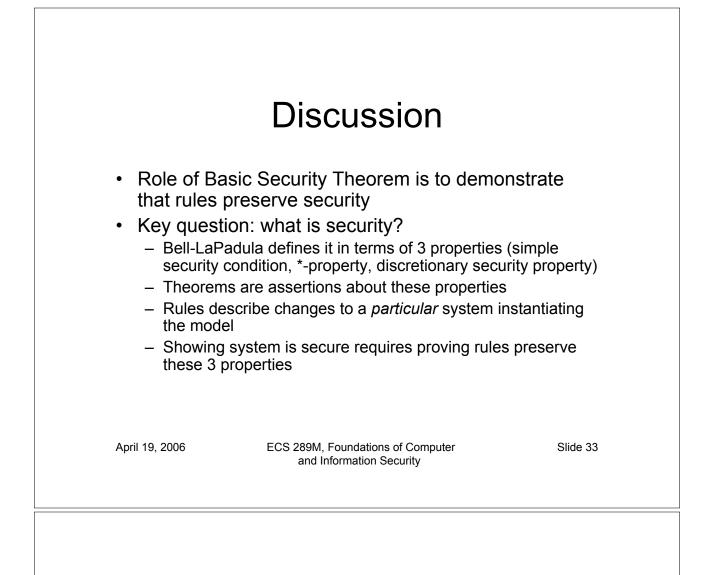
- Σ(R, D, W, z₀) satisfies the †-property relative to S'⊆ S for any secure state z₀ iff for every action (r, d, (b, m, f, h), (b', m', f', h')), W satisfies the following for every s ∈ S'
 - Every $(s, o, p) \in b b'$ satisfies the †-property relative to S'
 - Every (s, o, p) ∈ b' that does not satisfy the †-property relative to S' is not in b
- Σ(R, D, W, z₀) is a secure system if z₀ is a secure state and W satisfies the conditions for the simple security condition, the †-property, and the ds-property.

April 19, 2006

ECS 289M, Foundations of Computer and Information Security Slide 31

Problem

- This system is *clearly* non-secure!
 - Information flows from higher to lower because of the †-property



Rules and Model

- Nature of rules is irrelevant to model
- Model treats "security" as axiomatic
- · Policy defines "security"
 - This instantiates the model
 - Policy reflects the requirements of the systems
- McLean's definition differs from Bell-LaPadula
 ... and is not suitable for a confidentiality policy
- Analysts cannot prove "security" definition is appropriate through the model

System Z

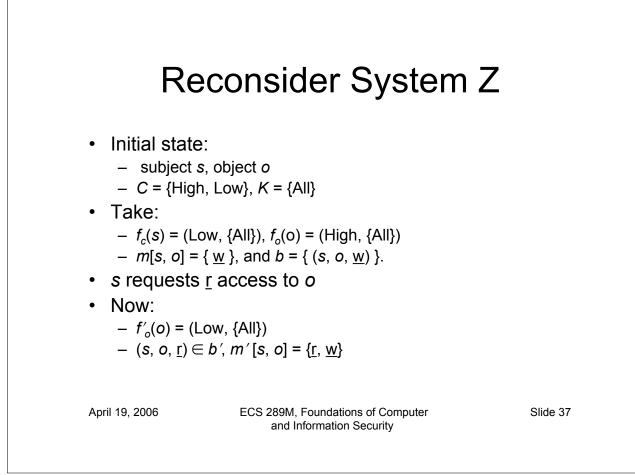
- · System supporting weak tranquility
- On any request, system downgrades all subjects and objects to lowest level and adds the requested access permission
 - Let initial state satisfy all 3 properties
 - Successive states also satisfy all 3 properties
- Clearly not secure
 - On first request, everyone can read everything

April 19, 2006

ECS 289M, Foundations of Computer and Information Security Slide 35

Reformulation of Secure Action

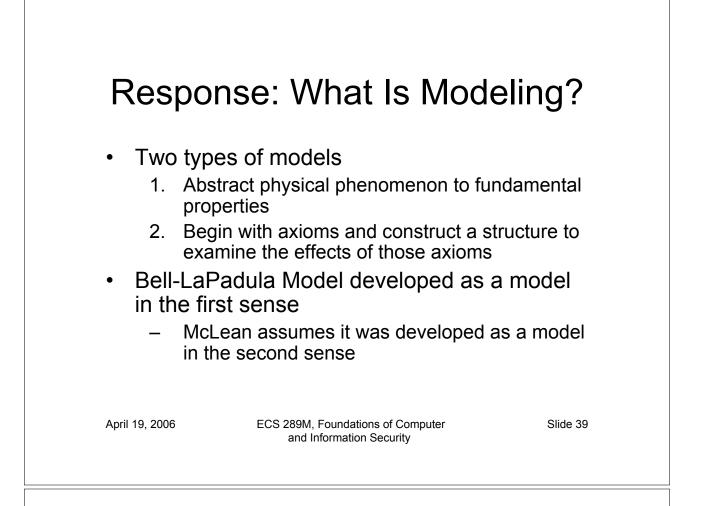
- Given state that satisfies the 3 properties, the action transforms the system into a state that satisfies these properties and eliminates any accesses present in the transformed state that would violate the property in the initial state, then the action is secure
- BST holds with these modified versions of the 3 properties



Non-Secure System Z

- As (s, o, <u>r</u>) ∈ b' b and f_o(o) dom f_c(s), access added that was illegal in previous state
 - Under the new version of the Basic Security Theorem, System Z is not secure
 - Under the old version of the Basic Security Theorem, as $f'_c(s) = f'_o(o)$, System Z is secure

April 19, 2006



Reconciling System Z

- Different definitions of security create different results
 - Under one (original definition in Bell-LaPadula Model), System Z is secure
 - Under other (McLean's definition), System Z is not secure