Experiment 2. We modify the configuration from Experiment 1 by applying Clulow's first suggestion: that attribute changing operations be prevented from allowing a stored key to have both wrap and decrypt set. Note that in order to do this, it is not sufficient merely to check that decrypt is unset before setting wrap, and to check wrap in unset before setting decrypt. One must also add wrap and decrypt to the list of sticky attributes which once set, may not be unset, or the attack is not prevented, [17]. Having applied these measures, we discovered a previously unknown attack, given in Figure 3. The intruder imports his own key k_3 by first encrypting it under k_2, and then unwrapping it. He can then export the sensitive key k_1 under k_3 to discover its value.

Initial state: The intruder knows the handles $h(n_1, k_1)$, $h(n_2, k_2)$ and the key k_3; n_1 has the attributes sensitive and extract set whereas n_2 has the attributes unwrap and encrypt set.

Trace:

SEncrypt:	$h(n_2, k_2), k_3$	$\rightarrow$	$senc(k_3, k_2)$
Unwrap:	$h(n_2, k_2), senc(k_3, k_2)$	$\xrightarrow{\text{new } n_3}$	$h(n_3, k_3)$
Set_wrap:	$h(n_3, k_3)$	$\rightarrow$	$wrap(n_3)$
Wrap:	$h(n_3, k_3), h(n_1, k_1)$	$\rightarrow$	$senc(k_1, k_3)$
Intruder:	$senc(k_1, k_3), k_3$	$\rightarrow$	k_1

Figure 3. Attack discovered in Experiment 2