Informelles Rahmenmodell für *Byzantine Agreement with Authentication*

Es gelten die Rahmenbedingungen aus Abschnitt 6.2.4, die im folgenden noch etwas präzisiert werden:

Zur Verwendung von Signaturen:
Signiert werden (im Unterschied zur Angabe im Buch) keine ganzen Nachrichten, sondern einzelne Werte oder bereits signierte Werte. (Dies entspricht der Konvention, dass eine externe Quelle zu Beginn signierte Werte verschickt.)
Jeder Prozess ist in der Lage, beliebige bereits signierte Werte mit seiner (einzigartigen) Signatur zu signieren. Auf diese Weise entsteht eine Schachtelung von Signaturen, deren Reihenfolge eindeutig ist. Das innerste Element dieser Schachtelung ist ein von der externen Quelle signierter Wert.
Als Notation verwenden wir hier $(v)_S$ für den von der Quelle S signierten Wert v und $((v)_S)_1$ für den von Prozess 1 signierten von der Quelle signierten Wert v, etc.
Jeder Prozess kann alle anderen Signaturen verifizieren, ist jedoch nicht in der Lage fremde Signaturen zu fälschen.

Der *EIGStop*-Algorithmus wird nicht geändert bis auf folgende kleine Anpassungen:
Nachrichten bestehen im originalen Algorithmus aus einzelnen Paaren (l, v) mit $v \in V$ und l ist das Label eines Knotens in einem EIG-Baum.
Jetzt sind die Paare von der Form (l, v_{sig}), l wieder ein Label, v_{sig} ist ein möglicherweise mehrfach signierter Wert $v_{sig} = (...((v)_S)_{i_1}...)_{i_k}$.
Eine Anpassung des Algorithmus an die Anforderungen der Authentifikation bedeutet, dass jeder Prozess i in der Lage ist, eine Nachricht (l, v_{sig}) von Prozess j als korrekt signiert einzuordnen, indem er

- das Label l auf korrekte Länge überprüft,
- anhand der äußersten Signatur des Wertes v_{sig} den Absender der Nachricht verifiziert
- verifiziert, dass die geschachtelten Signaturen tatsächlich von den entsprechenden Prozessen im Label stammen (implizit wird also auch die Reihenfolge der geschachtelten Signaturen mit der Reihenfolge der Prozess-IDs im Label verglichen)

Wird eine Nachricht so verifiziert wird der Wert v_{sig} dem Knoten mit dem Label lj zugeordnet. Entspricht eine Nachricht nicht diesen Kriterien, wird sie als gefälscht oder falsch signiert eingestuft. Eine solche Nachricht wird (entsprechend dem Byzantine failure model) als NULL gehandhabt und dem entsprechenden Knoten zugeordnet.
Sendet ein Prozess i einen Wert v_{sig}, der in seinem EIG-Baum am Knoten mit dem Label lj steht, so fügt er der Schachtelung von Signaturen seine eigene hinzu. Aus dem Ergebnis bildet er das Paar $(lj, (v_{sig})_i)$ und fügt es zur neuen Nachricht hinzu.

Konvention:
Zur Vereinfachung nehmen wir an, dass bei Vergleichsoperationen v_{sig} und v identifiziert werden, z.B. die Menge $\{v, (v)_S, ((v)_S)_1\}$ sei also (insbesondere für die Lemmata) einelementig.

Bei Unklarheiten in den Übungen fragen!