## Contents

| 1 | SM0r3Solutions Theory |             |   |
|---|-----------------------|-------------|---|
|   | 1.1                   | Definitions | 3 |
|   | 1.2                   | Theorems    | 3 |

## 1 SM0r3Solutions Theory

Built: 16 April 2019 Parent Theories: SM0r3

## 1.1 Definitions

```
[certificatesr3a_def]
 \vdash \forall npriv \ privemd \ cmd.
      certificatesr3a npriv privcmd cmd =
      MAP mkRCert
        (certsr1a npriv \ privcmd \ cmd \ ++
         certsr2root npriv privcmd) ++
      MAP (mkSCert (ca 0)) (certsr2signed npriv privcmd)
[certsr1a_def]
 \vdash \forall npriv \ privemd \ cmd.
      certsr1a npriv privcmd cmd =
      certs npriv privcmd ++
      [reps (Name (Staff Alice)) (Name (Role Commander))
         (prop (SOME cmd));
       reps (Name (Staff Bob)) (Name (Role Operator))
         (prop (SOME cmd))]
[certsr2a_def]
 \vdash \forall npriv \ privcmd \ cmd.
      certsr2a npriv privcmd cmd =
      certsr1a npriv privcmd cmd ++ certsr2root npriv privcmd ++
      certsr2signed npriv privcmd
1.2
      Theorems
[SMO_Commander_privcmd_trapped_lemma]
 \vdash CFGInterpret (M, Oi, Os)
      (CFG inputOK SMOStateInterp (certs npriv\ privcmd)
         (Name (Role Commander) says prop (SOME (PR privcmd))::
               ins) s outs) <math>\Rightarrow
    (M,Oi,Os) sat prop NONE
[SMO_Commander_trap_privcmd_justified_thm]
 \vdash \forall NS \ Out \ M \ Oi \ Os.
      TR (M, Oi, Os) (trap (PR privend))
        (CFG inputOK SMOStateInterp (certs npriv privcmd)
           (Name (Role Commander) says prop (SOME (PR privcmd))::
                 ins) s outs)
        (CFG inputOK SMOStateInterp (certs npriv privcmd) ins
           (NS \ s \ (trap \ (PR \ privemd)))
```

```
(Out \ s \ (trap \ (PR \ privemd))::outs)) \iff
     input0K
        (Name (Role Commander) says prop (SOME (PR privemd))) \land
     CFGInterpret (M, Oi, Os)
        (CFG inputOK SMOStateInterp (certs npriv privcmd)
           (Name (Role Commander) says prop (SOME (PR privcmd))::
                ins) s outs) \land (M, Oi, Os) sat prop NONE
[SMOr1_Commander_Alice_trap_privcmd_justified_thm]
 \vdash \forall NS \ Out \ M \ Oi \ Os.
     TR (M, Oi, Os) (trap (PR privcmd))
        (CFG inputOKr1 SMOStateInterp
           (certsr1a npriv privcmd (PR privcmd))
           (Name (Staff Alice) quoting Name (Role Commander) says
            prop (SOME (PR privcmd))::ins) s outs)
        (CFG inputOKr1 SMOStateInterp
           (certsr1a npriv privcmd (PR privcmd)) ins
           (NS \ s \ (trap \ (PR \ privemd)))
           (Out \ s \ (trap \ (PR \ privemd))::outs)) \iff
      inputOKr1
        (Name (Staff Alice) quoting Name (Role Commander) says
         prop (SOME (PR privcmd))) \cdot
     CFGInterpret (M, Oi, Os)
        (CFG inputOKr1 SMOStateInterp
           (certsr1a npriv privcmd (PR privcmd))
           (Name (Staff Alice) quoting Name (Role Commander) says
            prop (SOME (PR privend))::ins) s outs) \land
      (M, Oi, Os) sat prop NONE
SMOr1_Commander_mapSMOinputOperatorBob_trap_privcmd_justified_thm
 \vdash \forall s \ privem d \ outs \ npriv \ ins \ NS \ Out \ M \ Oi \ Os.
     TR (M, Oi, Os) (trap (PR privemd))
        (CFG inputOKr1 SMOStateInterp
           (certsr1a npriv privcmd (PR privcmd))
           (mapSMOinputOperatorBob
              (Name (Role Commander) says
               prop (SOME (PR privcmd)))::ins) s outs)
        (CFG inputOKr1 SMOStateInterp
           (certsr1a npriv privcmd (PR privcmd)) ins
           (NS \ s \ (trap \ (PR \ privemd)))
           (Out \ s \ (trap \ (PR \ privemd))::outs)) \iff
     inputOKr1
        (mapSMOinputOperatorBob
           (Name (Role Commander) says
            prop (SOME (PR privcmd)))) ∧
     CFGInterpret (M, Oi, Os)
        (CFG inputOKr1 SMOStateInterp
           (certsr1a npriv privcmd (PR privcmd))
           (mapSMOinputOperatorBob
```

## Index

```
{\bf SM0r3Solutions\ Theory},\,3
    Definitions, 3
      certificatesr3a_def, 3
      certsr1a\_def, 3
      certsr2a_def, 3
    Theorems, 3
      SM0\_Commander\_privcmd\_trapped\_-
        lemma, 3
      SM0\_Commander\_trap\_privcmd\_justi-
        fied_thm, 3
      SM0r1\_Commander\_Alice\_trap\_privcmd\_-
        justified_thm, 4
      SM0r1\_Commander\_mapSM0inputOperatorBob\_-
        trap\_privcmd\_justified\_thm, 4
      SM0r1\_mapSM0\_Alice\_Commander\_-
        trap\_privcmd\_lemma, 5
```