

SMART CONTRACT-BASED BLOCKCHAIN-ENVISIONED AUTHENTICATION SCHEME FOR SMART FARMING

ABSTRACT

A blockchain-based smart farming technology provides the agricultural data to the farmers and other users associated with smart farming on a single integrated platform. Moreover, persistence and auditability of stored data in blocks into the blockchain provide the confidence of using the correct data when needed later and adds transparency, anonymity and traceability at the same time. To fulfill such a goal, in this paper, we design a new smart contract-based blockchain-envisioned authenticated key agreement mechanism in a smart farming environment. The device-to-device (D2D) authentication phase and device-to-gateway (D2G) authentication phase support mutual authentication and key agreement between two Internet of Things (IoT) enabled devices and between an IoT device and the gateway node in the network, respectively. The blocks are created by the edge servers from the authenticated data of IoT devices received from the gateway nodes and then sent to the cloud server. The smart contract-based consensus mechanism allows verification and addition of the formed blocks by a Peer-to-Peer (P2P) cloud servers network. The security of the proposed scheme is done through formal and informal security analysis, and also using the formal security verification tool. A detailed comparative study reveals that the proposed scheme offers superior security and more functionality features as compared to existing authentication protocols. Finally, the blockchain based simulation has been conducted to measure computational time for a varied number of mined blocks and also a varied number of transactions per block.

REGISTRATION & AUTHENTICATION SCHEME

Trusted Registration Authority (RA)	IoT Smart Device (SN)
Pick $ID_S, TID_S, s_1 \in Z_q^*$ Compute $RID_S = H(ID_S s_1 mk_{RA})$ Pick $pr_S \in Z_q^*$ Compute $Pub_S = pr_S \cdot G$ $TC_S = H(RID_S pr_S mk_{RA} RTS_S)$ Preload SN with $\{(RID_S, TID_S, TC_S), H(\cdot), E_q(a, b), G, (pr_S, Pub_S)\}$	Store $\{(RID_S, TID_S, TC_S), H(\cdot), E_q(a, b), G, (pr_S, Pub_S)\}$ in its memory

Fig. 2: Summary of IoT smart device registration phase

Registration Authority (RA)	Gateway Node (GWN)	Edge Server (ES)	Cloud Server (CS)
Pick $TID_G, TID_C, s_1, ID_G, TID_G, e_1$ $ID_C, TID_C, s_1 \in Z_q^*$ Compute $RID_C = H(ID_C s_1 mk_{RA} RTS_C)$ $RID_G = H(ID_G s_1 mk_{RA} RTS_G)$ $RID_C = H(ID_C s_1 mk_{RA} RTS_C)$ Preload the gateway node GWN with $\{(RID_G, TID_G), (RID_S, TID_S, TC_S), H(\cdot), E_q(a, b), G\}$ Preload the edge server ES with $\{(RID_G, TID_G), (RID_S, TID_S, TC_S), H(\cdot), E_q(a, b), G\}$ Preload the cloud server CS with $\{(RID_C, TID_C), (RID_G), H(\cdot), E_q(a, b), G\}$	Pick $pr_G \in Z_q^*$ Compute $Pub_G = pr_G \cdot G$ Store the information $\{(RID_G, TID_G), (pr_G, Pub_G)\}$ in secure memory (database)	Pick $pr_E \in Z_q^*$ Compute $Pub_E = pr_E \cdot G$ Store the information $\{(RID_G, TID_G), (pr_E, Pub_E), H(\cdot), E_q(a, b), G\}$ in secure memory (database)	Pick $pr_C \in Z_q^*$ Compute $Pub_C = pr_C \cdot G$ Store the information $\{(RID_C, TID_C), (RID_G), (pr_C, Pub_C), H(\cdot), E_q(a, b), G\}$ in secure memory (database)

Fig. 3: Summary of registration phase of GWN, ES and CS

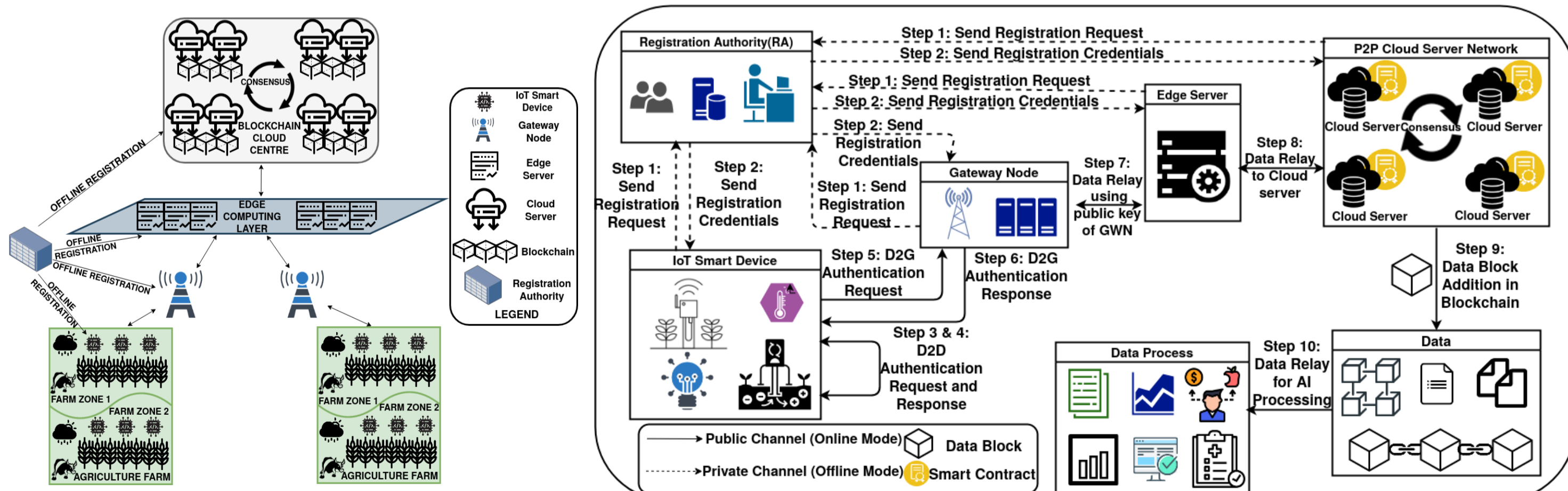
IoT Smart Device (SN ₁)	IoT Smart Device (SN ₂)
Pick $r_{S1} \in Z_q^*$ and timestamp T_{S1} Compute $x_{S1} = H(RID_{S1} TID_{S1} TC_{S1} pr_{S1} T_{S1} r_{S1})$ $X_{S1} = x_{S1} \cdot G$ Calculate $Sig_{S1} = x_{S1} + H(TID_{S1} Pub_{S1} T_{S1}) \cdot pr_{S1} \pmod{q}$ $Mag_{S1} = (TID_{S1}, X_{S1}, Sig_{S1}, Pub_{S1}, T_{S1})$ via public channel	Check if $ T_{S1} - T_{S2} \leq \Delta T$? If so, check if $Sig_{S1} \cdot G - X_{S1} + H(TID_{S1} Pub_{S1} T_{S1}) \cdot pr_{S1} \stackrel{?}{=} Pub_{S1}$? If so, generate $r_{S2} \in Z_q^*$, T_{S2} Compute $SK_{S1S2} = pr_{S2} \cdot X_{S1}$ Compute $Sig_{S2} = pr_{S2} + H(TID_{S2} T_{S2} Pub_{S2} SK_{S1S2} T_{S2}) \cdot pr_{S2} \pmod{q}$ $Mag_{S2} = (TID_{S2}, X_{S2}, Sig_{S2}, Pub_{S2}, T_{S2})$ via public channel
Check if $ T_{S2} - T_{S1} \leq \Delta T$? If so, compute $SK_{S1S2} = x_{S1} \cdot Y_{S2}$ $Sig_{S2} \cdot G - Y_{S2} + H(TID_{S2} T_{S2} Pub_{S2} SK_{S1S2} T_{S2}) \cdot pr_{S2} \stackrel{?}{=} Pub_{S2}$ Generate T_{S2} Compute $SK_{S1S2} = H(SK_{S1S2} T_{S2})$ $Mag_{S1S2} = (SK_{S1S2}, T_{S2})$ via public channel	Check if $ T_{S2} - T_{S1} \leq \Delta T$? If so, compute $SK_{S1S2} = H(SK_{S1S2} T_{S2})$ Check $SK_{S1S2} \stackrel{?}{=} SK_{S1S2}$ If so, store SK_{S1S2} Both SN_1 and SN_2 share the same secret key $SK_{S1S2} = SK_{S2S1}$

Fig. 4: Summary of D2D authentication phase

IoT Smart Device (SN)	Gateway Node (GWN)
Pick $pr \in Z_q^*$, timestamp T_S Calculate $A_S = H(TID_S pr T_S) \cdot G$ $x_S = H(pr pr RID_S T_S) \cdot pr \pmod{q}$ $Sig_S = H(TID_S pr T_S) + H(x_S TID_S A_S T_S TC_S) \cdot pr \pmod{q}$ $Mag_{S1} = (TID_S, A_S, x_S, Sig_S, T_S)$ via public channel	Check if $ T_S - T_{S1} \leq \Delta T$? If so, check if $Sig_S \cdot G = A_S + H(x_S TID_S A_S T_S TC_S) \cdot pr$? If so, generate $pr_G \in Z_q^*$, timestamp T_{S1} $E_S = H(TID_S pr RID_S T_S) \cdot G$ $DK_{GS} = H(TID_S pr RID_S T_S) \cdot A_S$ $pr_G = H(pr TID_S RID_S TC_S T_S) \cdot pr \pmod{q}$ $SK_{GS} = H(DK_{GS} pr H(pr TID_S RID_S TC_S T_S) T_{S1})$ generate $TID_G^* \in Z_q^*$ $TID_G^* = H(TID_S pr RID_S T_S) + H(TID_S pr RID_S TC_S T_S) \cdot pr \pmod{q}$ $SK_{GS} = H(DK_{GS} pr H(pr TID_S RID_S TC_S T_S) T_{S1})$ via public channel
Check if $ T_{S1} - T_S \leq \Delta T$? If so, compute $SK_{GS} = pr_G \cdot A_S$ $TID_G^* = H(DK_{GS} pr RID_S T_S) + H(TID_S pr RID_S TC_S T_S) \cdot pr \pmod{q}$ $SK_{GS} = H(DK_{GS} pr H(pr TID_S RID_S TC_S T_S) T_{S1})$ via open channel	Check if $ T_{S1} - T_S \leq \Delta T$? If so, compute $SK_{GS} = H(SK_{GS} TID_G^* T_{S1})$ Check if $SK_{GS} \stackrel{?}{=} SK_{GS}$ GWN also updates TID_G by TID_G^* in its secure database Both SN and GWN share the same secret key $SK_{GS} = SK_{GS}$

Fig. 5: Summary of D2G authentication phase

NETWORK MODEL & ACCESS MODEL



BLOCKCHAIN CONSENSUS ALGORITHM RESULTS

Algorithm 1 Smart contract processing and consensus workflow of the blockchain

Input: $Block_i$: A full block having the structure as given in Fig. 16 that is to be added to the blockchain, N : Total number of P2P nodes (cloud servers) in the blockchain network
Output: Block commit status (YES/NO)

- 1: Set $Magic_Number = 2 * (N - 1) / 3 + 1$
- 2: $CMP_i \leftarrow \Phi$ (empty)
- 3: Broadcast $Block_i$ to the replica nodes in the network to peers
- 4: for each replica cloud server node CS_j do
- 5: /* Smart contract processing */
- 6: Set $Consensus_Vote_j = NO$
- 7: Compute $Block_Hash = H(Block_i)$
- 8: if (Block Hash = Curr Block Hash) then
- 9: if (validation of $Sig_{paraBlock}$ using Pub_{ES} is successful) then
- 10: Generate Merkle tree root ($MTR_{i,x}$) using the n_i transactions stored in the block payload
- 11: if ($MTR_{i,x} = MTR_{rx}$) then
- 12: Set $Consensus_Vote_j = YES$
- 13: end if
- 14: end if
- 15: end if
- 16: Add $Consensus_Vote_j$ to CMP_i
- 17: end for
- 18: Set $Appcount \leftarrow 0$
- 19: for each vote V reply in CMP_i do
- 20: if (V is YES) then
- 21: Set $Appcount = Appcount + 1$
- 22: end if
- 23: end for
- 24: if ($Appcount \geq Magic_Number$) then
- 25: Add block $Block_i$ into the blockchain
- 26: Broadcast block commit status as YES to the blockchain network
- 27: end if

SUMMARY SAFE	SUMMARY SAFE
DETAILS BOUNDED_NUMBER_OF_SESSIONS TYPED_MODEL PROTOCOL /home/anusha/Desktop /span/testsuite/results/D2D.if GOAL As Specified BACKEND CL-AtSe STATISTICS Analysed : 179 states Reachable : 44 states Translation: 0.06 seconds Computation: 0.21 seconds	DETAILS BOUNDED_NUMBER_OF_SESSIONS PROTOCOL /home/anusha/Desktop /span/testsuite/results/D2D.if GOAL As specified BACKEND OFMC STATISTICS TIME 663 l ms parseTime 0 ms visitedNodes: 2960 nodes depth: 7 plies

Fig. 8: AVISPA simulation results for D2D authentication

SUMMARY SAFE	SUMMARY SAFE
DETAILS BOUNDED_NUMBER_OF_SESSIONS TYPED_MODEL PROTOCOL /home/anusha/Desktop /span/testsuite/results/auth.if GOAL As Specified BACKEND CL-AtSe STATISTICS Analysed : 7 states Reachable : 7 states Translation: 0.05 seconds Computation: 0.01 seconds	DETAILS BOUNDED_NUMBER_OF_SESSIONS PROTOCOL /home/anusha/Desktop /span/testsuite/results/auth.if GOAL As specified BACKEND OFMC STATISTICS TIME 630 ms parseTime 0 ms visitedNodes: 434 nodes depth: 7 plies

Fig. 9: AVISPA simulation results for D2G authentication






INTERNATIONAL INSTITUTE OF
INFORMATION TECHNOLOGY
HYDERABAD

R&D SHOWCASE 2021






INTERNATIONAL INSTITUTE OF
INFORMATION TECHNOLOGY
HYDERABAD

R&D SHOWCASE 2021






INTERNATIONAL INSTITUTE OF
INFORMATION TECHNOLOGY
HYDERABAD

R&D SHOWCASE 2021




INTERNATIONAL INSTITUTE OF
INFORMATION TECHNOLOGY
HYDERABAD

R&D SHOWCASE 2021