Algorithm 1 Ghost Backdoor based on neuron select

Input: central server C_s, a set of all client C, current epoch e, end epoch E_e, current client C_i, learning rate η, benign datasets D, mask matrix $\mathbb{R}^{r\times d}_{mask}$ ghost neurons' values matrix $\mathbb{R}^{r\times d}_{V_s}$

Output: a global model with high accuracy, ghost backdoor and high accuracy in main-task

C_s select n clients by random into C_n
C_s build a global model G
C_s send G to each client in C_n
choose the ghost neurons
pre-train with benign samples to collect the values of every neurons
choose V_s as trigger
for $e <\ E_e$ **do**
  for the k-th client C_e^k in C_n **do**
    Download G as local model L and train by D,
    Compute gradient by D on batch B_i of size ℓ
    $g_{e+1}^k = \frac{1}{\ell}\sum_{i=1}^{\ell} \nabla_\theta \mathcal{L}(\theta_{C_e^k}, D)$
    if client C_i is advisary **and** epoch mod N_{attack} = 0 **then**
      $\hat{g}_{e+1}^k = g_{e+1}^k * \mathbb{R}^{r\times d}_{mask} + \mathbb{R}^{r\times d}_{V_s}$
      Update $\theta_{C_{e+1}^k} = \theta_{C_e^k} - \eta \hat{g}_{e+1}^k$
    else
      Update $\theta_{C_{e+1}^k} = \theta_{C_e^k} - \eta g_{e+1}^k$
    Upload $\theta_{C_{e+1}^k}$ to C_s
  C_s recieve $\sum_1^k \theta_{C_{e+1}^k}$ and generate update U_{e+1} for G_{last}
  $G = G_{last} - U_{e+1}$
return Final global model G with backdoor